



Guía del usuario

# AWS DeepRacer Estudiante



# AWS DeepRacer Estudiante: Guía del usuario

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

---

# Table of Contents

¿Qué es AWS DeepRacer Student? .....	1
¿Le interesa el programa de becas de IA y ML de AWS? .....	2
¿Es la primera vez que utiliza AWS DeepRacer Student? .....	2
¿Le interesa competir en la Student League de AWS DeepRacer? .....	2
¿Qué es el programa de becas de IA y ML de AWS? .....	3
Lo que se gana con la beca de IA y ML de AWS .....	3
¿Cómo me registro y precalifico para el programa de becas de IA y ML de AWS? .....	4
Preguntas frecuentes (FAQ) sobre el programa de becas de IA y ML de AWS y la plataforma Udacity .....	5
¿Qué son las cuentas de AWS jugador? .....	8
Crear una cuenta de AWS jugador para los servicios compatibles .....	8
Eliminar una cuenta de AWS jugador .....	8
Introducción .....	10
Requisitos previos .....	10
Paso 1: Registrarse en AWS DeepRacer Student .....	10
Paso 2: Completar la inscripción para AWS DeepRacer Student .....	11
Paso 3: (opcional) Obtener información y solicitar la beca de IA y ML de AWS .....	12
Paso 4: Actualizar su perfil .....	12
Paso 5: Explorar AWS DeepRacer Student desde la página de Inicio .....	14
Entrene un modelo de aprendizaje por refuerzo .....	16
Paso 1: Entrenar un modelo de aprendizaje por refuerzo con AWS DeepRacer Student .....	16
Paso 2: Asignar un nombre al modelo .....	16
Paso 3: Elegir su pista .....	17
Paso 4: Elegir un algoritmo .....	17
Paso 5: Personalizar su función de recompensa .....	18
Paso 6: Elegir la duración y enviar su modelo a la tabla de clasificación .....	19
Paso 7: Ver el rendimiento de su modelo en la tabla de clasificación .....	20
Paso 8: Usar Clonar para mejorar su modelo .....	20
Paso 9: (opcional) Descargar un modelo .....	21
Únase a una carrera .....	22
Únase a una carrera de la Student League .....	22
Únase a una carrera comunitaria de estudiantes .....	23
Personalice una función de recompensa .....	24
Editar el código Python para personalizar su función de recompensa .....	24

Parámetros de entrada de la función de recompensa .....	25
all_wheels_on_track .....	26
closest_waypoints .....	28
closest_objects .....	30
distance_from_center .....	30
heading .....	32
is_crashed .....	32
is_left_of_center .....	32
is_offtrack .....	33
is_reversed .....	33
objects_distance .....	33
objects_heading .....	34
objects_left_of_center .....	34
objects_location .....	34
objects_speed .....	34
progress .....	35
speed .....	35
steering_angle .....	36
pasos .....	37
track_length .....	38
track_width .....	38
x, y .....	39
waypoints .....	40
Seguridad .....	42
Protección de los datos .....	42
Datos almacenados en el portal AWS DeepRacer Student .....	43
El cifrado en reposo en el portal AWS DeepRacer Student .....	43
Cifrado en tránsito en el portal AWS DeepRacer Student .....	44
Administración de identidades y accesos .....	44
Validación de conformidad .....	44
Resiliencia .....	45
Seguridad de la infraestructura .....	45
Solución de problemas comunes .....	46
¿Por qué se cerró sesión automáticamente en mi cuenta de AWS DeepRacer estudiante? .....	46
¿Cómo puedo dejar de participar en el programa de becas de IA y ML de AWS? .....	46
No puedo eliminar mi cuenta AWS DeepRacer de estudiante .....	47

---

No encuentro el nombre de mi escuela en la lista desplegable .....	47
No puedo seguir entrenando a mi modelo .....	47
Aparece el mensaje de error “Hay una cuenta registrada con este correo electrónico” .....	47
Me he registrado con una cuenta de Gmail y no encuentro mi código de verificación .....	48
Cuotas .....	49
Eliminación de la cuenta .....	50
Historial de documentos .....	52
.....	liii

# ¿Qué es AWS DeepRacer Student?

AWS DeepRacer Student es un lugar para que los estudiantes matriculados en escuelas secundarias y universidades de todo el mundo desarrollen habilidades de machine learning (ML). Proporciona acceso a material educativo, al programa opcional de becas IA y ML de AWS y la oportunidad de entrenar y probar modelos de aprendizaje por refuerzo (RL) para la AWS DeepRacer Student League. Para comenzar, consulte los temas de la sección [???](#).

## Características de AWS DeepRacer Student

- **Inicio:** encuentre detalles sobre los próximos eventos, practique el entrenamiento de modelos RL, acceda a los contenidos educativos de ML y realice un seguimiento de sus horas de entrenamiento de modelos. También puede gestionar su perfil de AWS DeepRacer Student y la información de su cuenta desde el panel de navegación izquierdo.
- **Aprender:** acceda al contenido de ML, incluidos los vídeos, desarrollados por expertos de AWS. Los estudiantes sin experiencia previa pueden aprender los fundamentos del ML o a partir de un material fácil de entender y a su propio ritmo.
- **Practicar:** elija una pista, un algoritmo y una función de recompensa para crear un modelo de RL. Si lo desea, realice un recorrido guiado por el código Python de la función de recompensa y elija personalizarlo. Entrene su modelo en un entorno de carreras 3D simulado con el servicio AWS DeepRacer. Clone sus modelos con mejor rendimiento y modifique sus funciones de recompensa para ascender en la clasificación de la Student League de AWS DeepRacer.
- **Competir:** envíe sus modelos a carreras virtuales mensuales para conseguir logros y la oportunidad de ganar premios. Los mejores corredores estudiantiles de la región tienen la oportunidad de competir en el Campeonato Mundial Estudiantil. Para obtener más información, consulte los [Términos y condiciones](#). Todos los estudiantes tienen 10 horas al mes para entrenar modelos RL para la Student League de AWS DeepRacer. Solo los modelos entrenados con AWS DeepRacer Student pueden usarse en la AWS DeepRacer Student League.

## Temas

- [¿Le interesa el programa de becas de IA y ML de AWS?](#)
- [¿Es la primera vez que utiliza AWS DeepRacer Student?](#)
- [¿Le interesa competir en la Student League de AWS DeepRacer?](#)

## ¿Le interesa el programa de becas de IA y ML de AWS?

Si se autoidentifica como persona carente de recursos o de representación necesarios en el ámbito tecnológico, opte por el programa de becas de IA y ML de AWS. Para obtener más información sobre los requisitos, cómo postularse y qué puede ganar, consulte [¿Qué es el programa de becas de IA y ML de AWS?](#)

## ¿Es la primera vez que utiliza AWS DeepRacer Student?

Si es la primera vez que utiliza AWS DeepRacer Student, le recomendamos que empiece leyendo las siguientes secciones:

- [Introducción a AWS DeepRacer Student](#)
- [Únase a una carrera de AWS DeepRacer Student](#)
- [Entrenamiento de un modelo de aprendizaje por refuerzo en AWS DeepRacer Student](#)

## ¿Le interesa competir en la Student League de AWS DeepRacer?

La Student League de AWS DeepRacer es una competición patrocinada por AWS en la que los estudiantes pueden competir en carreras virtuales mensuales entre marzo y septiembre para competir a nivel nacional y regional y clasificarse para tener la oportunidad de competir en el Campeonato Mundial Estudiantil de octubre. Los estudiantes tienen la oportunidad de ganar recompensas y premios. Para obtener más información, consulte los [Términos y condiciones](#).

Para competir en su primera carrera, consulte [Únase a una carrera de AWS DeepRacer Student](#).

# ¿Qué es el programa de becas de IA y ML de AWS?

El programa de becas de IA y ML de AWS, que se lanzó como parte de AWS DeepRacer Student, está diseñado para llevar la diversidad al campo de la inteligencia artificial (IA) y el machine learning (ML) al ofrecer a los candidatos seleccionados la oportunidad de obtener hasta dos nanogrados de Udacity. Un nanogrado de Udacity es un curso masivo y abierto en línea (MOOC) diseñado para cerrar la brecha entre los objetivos de aprendizaje y los objetivos profesionales. Para obtener más información, consulte [¿Qué es un programa de nanogrados?](#) en la documentación de soporte de Udacity. Los candidatos seleccionados también tendrán acceso a eventos exclusivos y a tutorías que les ayudarán a avanzar en sus carreras. Para obtener más información, consulte [Lo que se gana con la beca de IA y ML de AWS](#).

Esta beca va dirigida a personas con pocos recursos y representación en el sector tecnológico. Los solicitantes deben tener al menos 16 años y estar matriculados en una escuela secundaria, universidad o centro de formación profesional. Para obtener más información sobre el modo en que puede precalificar, consulte [¿Cómo me registro y precalifico para el programa de becas de IA y ML de AWS?](#)

La participación en el programa de becas de IA y ML de AWS es gratuita.

El programa de becas de IA y ML de AWS se lanzó oficialmente el 11 de abril de 2022.

El programa de becas de IA y ML de AWS se basa en un enfoque basado en grupos. Cada año, se seleccionan dos grupos de 1000 estudiantes (2000 en total). Las solicitudes para el primer grupo se enviarán a Udacity el 31 de mayo de 2024 y las solicitudes para el segundo, el 30 de septiembre de 2024.

## Lo que se gana con la beca de IA y ML de AWS

Para el programa de becas de IA y ML de AWS, AWS colabora con Udacity.

Cada año, el programa de becas de IA y ML de AWS ofrecerá en todo el mundo 2500 becas de nanogrado de Udacity, distribuidas en dos grupos diferentes. Todos los estudiantes reciben lo siguiente:

- Entrada gratuita al nanogrado de Programación de IA con Python. Este curso enseña las habilidades fundamentales necesarias para empezar a utilizar técnicas de IA y desarrollar sus habilidades en programación, álgebra lineal y redes neuronales.

- Acceso al horario de oficina con los profesores de Udacity durante la semana para ayudarle a responder preguntas sobre el contenido de la clase. Además, los estudiantes pueden participar en ejercicios de casos de estudio semanales dirigidos por los profesores de Udacity.
- Acceso exclusivo a recursos profesionales y de mentoría, que incluyen sesiones con expertos del sector, orientación sobre la edición de currículum y consejos para entrevistas.

Además, los 500 mejores estudiantes de la clase de Programación de IA con Python (según el rendimiento en las evaluaciones del curso) tienen acceso a un nanogrado más avanzado que cubre los fundamentos del aprendizaje profundo y les proporciona las habilidades necesarias para ser ingenieros de machine learning.

Todos los estudiantes que cumplan los requisitos de la beca y presenten una solicitud de Udacity tendrán acceso gratuito a un nuevo curso de IA generativa, titulado Introducción a la IA generativa con AWS, creado conjuntamente por AWS y Udacity. El curso, de un mes de duración, es un recurso opcional para ayudar a los estudiantes a mejorar sus habilidades en el campo de la IA generativa. El curso está disponible para todos los estudiantes que presenten una solicitud para la beca, mientras que la beca de nanogrado solo está disponible para las personas becadas.

## ¿Cómo me registro y precalifico para el programa de becas de IA y ML de AWS?

El programa de becas de IA y ML de AWS está dirigido a estudiantes carentes de recursos o de representación necesarios que tengan 16 años o más. Entre los estudiantes que carecen de recursos y de representación necesarios se encuentran (entre otros) las mujeres, las personas con discapacidad, las personas de color (negras, latinas e indígenas) y los miembros de la comunidad LGBTQ+.

Para empezar, los estudiantes deben registrarse y crear una cuenta de AWS DeepRacer Student. Al inscribirse, puede optar por el programa de becas de IA y ML de AWS marcando la casilla de verificación. Cuando su cuenta esté configurada, verá un icono en la página de Inicio para hacer un seguimiento de las tareas de precalificación de su solicitud de beca. Los estudiantes deben completar dos requisitos previos en AWS DeepRacer Student para presentar su solicitud para el programa de becas.

Los estudiantes que precalifiquen para la beca de IA y ML de AWS recibirán un código de acceso único que podrán canjear en el sitio de Udacity para acceder al formulario de solicitud de beca de nanogrado de Udacity. Los estudiantes que precalifiquen no tienen garantizadas las becas de

nanogrado de Udacity. Udacity determina qué estudiantes precalificados reciben las becas de nanogrado de Udacity.

La precalificación para la beca se basa en dos criterios:

Revisar el trabajo del curso y aprobar las evaluaciones

Para precalificar, se debe obtener una puntuación del 80% o más en todas las evaluaciones obligatorias. Cada evaluación se basa en distintos capítulos que puede encontrar en la sección Aprender del panel de navegación de AWS DeepRacer Student. La solicitud de becas se abre el 1 de febrero de 2024.

Consigue un tiempo de vuelta mínimo

Cada mes, de febrero a septiembre, se lanza una nueva clasificación en la DeepRacer League de AWS, con un nuevo circuito. Para poder optar por la beca, tendrá que conseguir un tiempo inferior a dos minutos en una sola vuelta en una carrera individual.

Una vez que los estudiantes completen ambos requisitos previos, recibirán un correo electrónico de AWS con un código de acceso único y un enlace a la solicitud de la beca de nanogrado de Udacity. Para ser considerados para la beca, los estudiantes deben completar la solicitud en su totalidad. Cuando la solicitud esté completa, Udacity contacta a los estudiantes con los pasos necesarios para inscribirse en el curso opcional Introducción a la IA generativa de AWS que se ofrece a todos los candidatos. Udacity contacta a los candidatos que hayan sido seleccionados para recibir una beca de nanogrado de Udacity.

## Preguntas frecuentes (FAQ) sobre el programa de becas de IA y ML de AWS y la plataforma Udacity

¿Cuáles son los términos y condiciones oficiales de las becas de IA y ML de AWS?

Para ver los términos y condiciones oficiales, consulte los [términos y condiciones oficiales del programa de becas de IA y ML de AWS](#).

¿A quién va dirigida esta beca?

El programa de becas de IA y ML de AWS está dirigido a estudiantes de todo el mundo de 16 años o más que carezcan de los recursos y la representación necesarios. Entre los estudiantes que carecen de recursos y de representación necesarios se encuentran (entre otros) las mujeres, las

personas con discapacidad, las personas de color (negras, latinas e indígenas) y los miembros de la comunidad LGBTQ+.

¿Qué es un nanogrado?

Un nanogrado es un programa educativo en línea basado en habilidades que ayuda a cerrar la brecha entre el aprendizaje y las aptitudes profesionales.

Si precalifico para la beca de IA y ML de AWS, ¿recibiré automáticamente una beca de nanogrado de Udacity?

No. Los estudiantes que precalifiquen para la beca de IA y ML de AWS solo tienen acceso a la solicitud de la beca de nanogrado de Udacity. Udacity determina qué estudiantes precalificados reciben las becas de nanogrado de Udacity.

¿Debo presentar un comprobante de inscripción antes de ser seleccionado para la beca de IA y ML de AWS?

Es posible que se le pida que presente un comprobante de inscripción (como un expediente académico universitario) para recibir la beca de IA y ML de AWS.

¿Cómo indico que actualmente estoy matriculado en la escuela secundaria al solicitar el programa de becas de IA y ML de AWS?

Para indicar que está matriculado en la escuela secundaria, consulte [Paso 4: Actualizar su perfil](#) en la Guía de usuario de AWS DeepRacer Student.

¿Cómo se protegen los datos de mi información de identificación personal (PII)?

Sus datos están protegidos en la nube de AWS. Para obtener más información, consulte el [Aviso de privacidad de AWS](#).

¿Hay algún requisito de edad para solicitar la beca de IA y ML de AWS?

Sí. Para solicitar el programa de becas de IA y ML de AWS, debe tener al menos 16 años cuando se inscriba.

¿Cómo se me notificará si recibo la beca de IA y ML de AWS?

Recibirá un correo electrónico de Udacity sobre el estado de su solicitud de beca de nanogrado una vez que haya transcurrido el plazo de presentación de solicitudes.

¿Cuándo se me notificará si recibo la beca de IA y ML de AWS?

Udacity se pondrá en contacto con los solicitantes para informarles sobre el estado de su beca. Los estudiantes que sean seleccionados para una beca recibirán información sobre las fechas importantes.

¿Cómo puedo acceder al curso de IA generativa de AWS en Udacity?

Los estudiantes deben precalificar en AWS DeepRacer Student y presentar una solicitud de beca en Udacity. Udacity se pone en contacto con los estudiantes después de que hayan presentado las solicitudes completas y les proporciona los pasos para inscribirse en el curso de Introducción a la IA generativa con AWS.

¿Qué ocurre si me aceptan en el programa de becas de IA y ML de AWS?

Una vez que se le notifique que ha recibido la beca de IA y ML de AWS, un representante del equipo de incorporación de Udacity se pondrá en contacto con usted mediante el correo electrónico que proporcionó al completar su solicitud.

Si recibo una beca de nanogrado de Udacity, ¿cuánto tiempo debo dedicar a mis estudios de nanogrado?

Los estudiantes deberían dedicar aproximadamente 10 horas a la semana a sus estudios de nanogradados en Udacity.

¿Se requiere dominio del idioma inglés?

Aunque no se requiere dominio del inglés, los materiales de aprendizaje están en inglés. Para lograr su aprendizaje, los estudiantes deben tener buenas habilidades de lectura y escritura en inglés.

¿Qué sucede si no encuentro las opciones con las que me identifico en la lista de raza o género?

En ambos casos, puede usar la opción Prefiero identificarme como (selecciona para escribir).

## ¿Qué son las cuentas de AWS jugador?

AWS Las cuentas de jugador son una solución de identidad gestionada para AWS DeepRacer múltiples usuarios y AWS DeepRacer estudiantes creada por AWS. Tu cuenta de AWS jugador contiene todos los recursos creados en cada uno de estos AWS servicios.

## Crear una cuenta de AWS jugador para los servicios compatibles

Al crear una cuenta para [AWS DeepRacer varios usuarios](#) o para [AWS DeepRacer estudiantes](#), se crea automáticamente una cuenta de AWS jugador. Cuando utilizas diferentes funciones de estos servicios, los nuevos recursos se añaden automáticamente a tu cuenta de AWS jugador. Para empezar con las funciones AWS DeepRacer multiusuario y AWS DeepRacer Student, utiliza los siguientes enlaces.

### Crear una cuenta de AWS DeepRacer estudiante

Para usar AWS DeepRacer Student, comience por crear una cuenta. Para obtener información sobre cómo crear una cuenta, consulte [Paso 1: Registrarse en AWS DeepRacer Student](#) la Guía del usuario para AWS DeepRacer estudiantes.

Utilice el AWS DeepRacer modo multiusuario para patrocinar a varios participantes en una sola cuenta.

AWS DeepRacer El modo multiusuario admite dos perfiles de usuario diferentes, administrador y participante. Ambos tienen requisitos de configuración diferentes. Para empezar, consulta el [modo multiusuario](#) en la Guía para AWS DeepRacer desarrolladores.

## Eliminar una cuenta de AWS jugador

Si eliminas una cuenta de AWS jugador, perderás inmediatamente el acceso a todos los servicios compatibles. Esto incluye todos los logros (insignias, puntos, avatares, etc.) que haya obtenido.

Al eliminar la cuenta de su cuenta de AWS jugador, no se elimina su AWS cuenta. Si también quieres eliminar tu AWS cuenta, sigue los pasos que se describen en [Cerrar tu AWS cuenta](#).

Si has utilizado tu cuenta de AWS jugador para crear un evento en modo AWS DeepRacer multiusuario, no podrás eliminar tu cuenta de AWS jugador. Esto es para garantizar que no se interrumpa la experiencia de los participantes en los eventos que haya creado. Para obtener más

información sobre cómo un administrador crea eventos en el modo AWS DeepRacer multiusuario, consulta el siguiente tema.

Configuración de eventos mediante el modo AWS DeepRacer multiusuario (administrador)

Para obtener información sobre cómo crear eventos mediante el modo multiusuario, consulte [Configurar el modo multiusuario \(administrador\)](#) en la AWS DeepRacer Guía para desarrolladores.

AWS Las cuentas de los jugadores no tienen acceso a ningún AWS recurso distinto de los creados en la cuenta del servicio. Todas AWS Identity and Access Management las políticas y los recursos asociados de la cuenta de servicio se limitan únicamente a los recursos necesarios.

# Introducción a AWS DeepRacer Student

Utilice esta explicación para comenzar con AWS DeepRacer Student. El tutorial explica cómo iniciar sesión en AWS DeepRacer Student, actualizar su perfil, solicitar la beca de IA y ML de AWS, hacer cursos gratuitos de machine learning (ML) y aprendizaje por refuerzo (RL) y crear modelos de AWS DeepRacer. Si no solicita la beca durante la creación de la cuenta, podrá hacerlo más adelante, cuando actualice su perfil.

## Temas

- [Requisitos previos](#)
- [Paso 1: Registrarse en AWS DeepRacer Student](#)
- [Paso 2: Completar la inscripción para AWS DeepRacer Student](#)
- [Paso 3: \(opcional\) Obtener información y solicitar la beca de IA y ML de AWS](#)
- [Paso 4: Actualizar su perfil](#)
- [Paso 5: Explorar AWS DeepRacer Student desde la página de Inicio](#)

## Requisitos previos

Para acceder a AWS DeepRacer Student y participar, necesita:

- Ser un estudiante que tenga al menos 16 años y que actualmente esté matriculado en una escuela secundaria, centro de formación profesional o universidad;
- o ser un educador u organizador de eventos para estudiantes de secundaria, universidad o centro de formación profesional.
- Dirección de correo electrónico válida.

## Paso 1: Registrarse en AWS DeepRacer Student

Puede registrarse en AWS DeepRacer Student mediante la URL que se proporciona en este procedimiento. Al registrarse, creará una cuenta de AWS jugador. Esta cuenta se puede utilizar con algunos otros servicios de AWS. Si ya tiene una cuenta de AWS jugador, puede utilizarla con AWS DeepRacer Student.

1. Abra la página de inicio <http://deepracerstudent.com/>.

2. Elija Comenzar.
3. En la página de inicio de sesión, si aún no tiene una cuenta de AWS jugador, seleccione Registrarse.

 Note

Si ya tiene una cuenta de AWS jugador, ingrese su información aquí. Para obtener más información acerca de la cuenta de AWS jugador, consulte [¿Qué son las cuentas de AWS jugador?](#)

4. En la página Registrarse, especifique la siguiente información:
  - Dirección de correo electrónico
  - Contraseña
5. Elija Registrarse. Se enviará un correo electrónico de confirmación a la dirección especificada.
6. En la ventana emergente que aparece, introduzca su número de verificación y seleccione Verificar.
7. En la página de Inicio de sesión de AWS DeepRacer Student, ingrese su dirección de correo electrónico y contraseña y seleccione Iniciar sesión.
8. En la ventana emergente Bienvenido a AWS DeepRacer Student, seleccione Completar registro.

 Note

Puede elegir Lo haré más tarde. Cierre sesión ahora si quiere registrarse más adelante.

## Paso 2: Completar la inscripción para AWS DeepRacer Student

Complete la sección para crear su cuenta de AWS DeepRacer Student. Todos los campos son obligatorios a menos que se indique lo contrario.

1. Rellene los campos de la sección Añada su información personal para crear su cuenta de AWS DeepRacer Student para crear su cuenta.
2. Seleccione la casilla de verificación para certificar que es un estudiante matriculado en una escuela secundaria, una universidad o un centro de formación profesional.

**Note**

Si es un estudiante de escuela secundaria, haga lo siguiente:

- En Escuela, seleccione Otro. Luego, agregue el nombre de su escuela secundaria al campo Ingrese el nombre de su escuela.
- Para la especialización actual o futura, elija Indeciso o elija una posible especialización de la lista.

## Paso 3: (opcional) Obtener información y solicitar la beca de IA y ML de AWS

Lea la información sobre la beca de IA y ML de AWS y a quién va destinada en la sección ¿Desea que se le considere para el programa de becas de IA y ML de AWS?.

1. Si cumple con los requisitos para la beca de IA y ML de AWS, puede solicitarla. Seleccione la casilla de verificación para confirmar que desea que se le considere para obtener una beca.

**Note**

Si no quiere que se le considere para una beca o está indeciso, deje la casilla en blanco y seleccione Lo haré más adelante.. Cierre sesión por ahora para cerrar sesión temporalmente o Enviar para continuar sin registrarse. También puede suscribirse al actualizar su perfil.

2. (Opcional) Utilice las listas desplegadas para introducir su información en los campos Elija sexo y Elija raza (solo para participantes de EE. UU.).
3. Elija Enviar.

## Paso 4: Actualizar su perfil

Para actualizar su perfil, vaya a la página Su perfil. También puede optar por inscribirse en el programa de becas de IA y ML de AWS. Si recibe logros por su actuación en la Student League de AWS DeepRacer, puede compartirlos en las redes sociales desde la página Su perfil.

Para actualizar su perfil:

1. En el panel de navegación de la izquierda de AWS DeepRacer Student, elija Su perfil.
2. En la página Su perfil, en Nombre del corredor, seleccione Cambiar su nombre de corredor.

 Note

Su nombre de corredor puede tener entre 2 y 24 caracteres. Se permiten letras, números y guiones (-).

3. En el campo Nombre del corredor, introduzca su nombre de corredor y seleccione Guardar. Si decide que no desea cambiar su nombre de corredor, seleccione Cancelar.
4. En la sección Información de su perfil, puede realizar cambios en los siguientes campos:
  - Nombre
  - Nombre del corredor
  - Nombre de la escuela
  - Nombre de la especialización
  - Año de graduación
  - País de residencia

 Note

Es necesario seleccionar un país de residencia para poder participar en las carreras virtuales de la Student League de AWS DeepRacer y recibir premios. Una vez que envíe su modelo a una carrera, su país de residencia quedará bloqueado para la temporada de carreras.

5. (Opcional) En la sección ¿Quiere que lo consideren para el programa de becas de IA y ML de AWS?, encontrará información sobre la beca y la forma de solicitarla.
  - a. Seleccione la casilla de verificación para optar por que se le considere para el programa de becas.
  - b. (Opcional) Ingrese su información en los campos Elegir género y Elegir raza.
  - c. Elija Enviar.

## Paso 5: Explorar AWS DeepRacer Student desde la página de Inicio

La página de Inicio de AWS DeepRacer Student es el lugar perfecto para empezar a explorar todo lo que AWS DeepRacer Student tiene para ofrecerle. Desde la página de Inicio puede hacer lo siguiente:

Comenzar a aprender los fundamentos del machine learning (ML)

Puede utilizar los cursos gratuitos disponibles en la sección Aprendizaje de AWS DeepRacer Student. Esta sólida oferta le ayuda a sentar las bases para su recorrido de machine learning con AWS DeepRacer Student.

Practique utilizando sus conocimientos de machine learning

Después de pasar algún tiempo utilizando los cursos de Aprendizaje, estará listo para crear y entrenar un modelo de AWS DeepRacer. Para obtener más información, consulte [Entrenamiento de un modelo de aprendizaje por refuerzo en AWS DeepRacer Student](#).

Compita en las carreras de AWS DeepRacer Student

Cuando haya terminado de entrenar a su primer modelo de AWS DeepRacer, estará listo para participar en una carrera. Si compite en una carrera mensual de una liga estudiantil, diríjase a la sección Carreras abiertas y seleccione Participar en la carrera. Cuando haya terminado de competir, compruebe el rendimiento de su modelo en la clasificación y los datos que necesita para realizar cambios y mejorar su modelo. Ganará puntos cuando compita en carreras virtuales mensuales. Estos puntos determinan su clasificación de la temporada nacional y regional. Los tres mejores corredores de cada mes ganarán medallas y premios. Para obtener más información, consulte [Únase a una carrera de AWS DeepRacer Student](#).

Consulte los horarios de entrenamiento del modelo

Cuando entrena y clona modelos, utiliza una parte del tiempo gratis de entrenamiento de modelos. Puede consultar las horas de entrenamiento restantes de sus modelos en la página de inicio.

Consulte el calendario de carreras de AWS DeepRacer Student

Consulte el calendario de carreras y comience a planificar su día de carrera.

Más información sobre la beca de IA y ML de AWS

Puede obtener información sobre la beca de IA y ML de AWS y lo que puede hacer para prepararse. Para obtener más información, consulte [¿Qué es el programa de becas de IA y ML de AWS?](#)

#### Ver otros recursos

Puede descubrir otros recursos que pueden ayudarlo a explorar AWS DeepRacer Student, como el canal Discord y el sitio web de AWS DeepRacer. Estos recursos le ayudan a conectarse con una comunidad de corredores y aficionados que comparten consejos e ideas.

# Entrenamiento de un modelo de aprendizaje por refuerzo en AWS DeepRacer Student

Este tutorial muestra cómo entrenar su primer modelo en AWS DeepRacer Student. También le proporciona algunos consejos útiles que le ayudarán a aprovechar al máximo su experiencia y acelerar su aprendizaje.

## Paso 1: Entrenar un modelo de aprendizaje por refuerzo con AWS DeepRacer Student

Comience su viaje en AWS DeepRacer Student aprendiendo dónde encontrar el botón Crear modelo y comience a entrenar su primer modelo. Tenga en cuenta que crear y entrenar un modelo es un proceso repetitivo. Experimente con diferentes algoritmos y funciones de recompensa para lograr los mejores resultados.

Entrene un modelo de aprendizaje por refuerzo

1. En la página de Inicio de AWS DeepRacer, seleccione Crear un modelo. Como alternativa, vaya a Sus modelos en el panel de navegación en la parte izquierda. En la página Modelos en Sus modelos, seleccione Crear modelo.
2. En la página de Descripción general, obtenga más información acerca de cómo entrenar un modelo de refuerzo. En esta página se explica cada paso del proceso. Cuando haya terminado de leerlo, elija Siguiente.

## Paso 2: Asignar un nombre al modelo

Póngale un nombre a su modelo. Es una buena práctica dar a los modelos nombres únicos para localizarlos rápidamente cuando desee mejorarlos o clonarlos. Por ejemplo, es posible que desee asignar un nombre a sus modelos utilizando una convención de nomenclatura como: *susiniciales-fecha-versión*.

Póngale un nombre a su modelo

1. En la página Asignar un nombre a su modelo, introduzca un nombre en el campo Nombre del modelo.

**Note**

Al empezar a entrenar un modelo, el nombre del modelo pasa a ser fijo y ya no se puede cambiar.

2. Elija Siguiente.

## Paso 3: Elegir su pista

Elija su pista de simulación. La pista sirve como entorno y proporciona datos a su vehículo. Si elige una pista muy compleja, su vehículo requerirá un tiempo total de entrenamiento más prolongado y la función de recompensa que utilizará es más compleja.

Elija su pista (entorno)

1. En la página Elegir pista, elija una pista que sirva de entorno de entrenamiento para su vehículo.
2. Elija Siguiente.

## Paso 4: Elegir un algoritmo

El AWS DeepRacer Student tiene dos algoritmos de entrenamiento entre los que puede elegir. Los diferentes algoritmos maximizan las recompensas de distintas formas. Para aprovechar al máximo su experiencia con AWS DeepRacer Student, experimente con ambos algoritmos. Para obtener más información sobre los algoritmos, consulte los [algoritmos de entrenamiento de AWS DeepRacer](#).

Elija un algoritmo de entrenamiento

1. En la página Elegir el tipo de algoritmo, seleccione un tipo de algoritmo. Hay dos tipos de algoritmos disponibles:
  - Optimización de políticas proximales (PPO) Este algoritmo estable pero hambriento de datos funciona de manera consistente entre las repeticiones de entrenamiento.
  - Soft Actor Critic (SAC) Este algoritmo inestable pero eficiente en términos de datos puede funcionar de manera inconsistente entre las repeticiones de entrenamiento.
2. Elija Siguiente.

## Paso 5: Personalizar su función de recompensa

La función de recompensa es una parte fundamental del aprendizaje por refuerzo. Úsela para incentivar a su vehículo (agente) a realizar acciones específicas mientras explora la pista (entorno). Del mismo modo que estimularía y desalentaría ciertos comportamientos en una mascota, puede utilizar esta herramienta para animar a su vehículo a completar una vuelta lo más rápido posible y evitar que se salga de la pista y haga zigzag.

Cuando entrene a su primer modelo, quizás quiera usar una función de recompensa de muestra predeterminada. Cuando esté listo para probar y optimizar su modelo, puede personalizar la función de recompensa editando el código en el editor de códigos. Para obtener más información acerca de cómo personalizar una función de recompensa, consulte [Personalizar una función de recompensa](#).

Para personalizar su función de recompensa:

1. En la página Personalizar la función de recompensa, seleccione un ejemplo de función de recompensa. Hay tres ejemplos de funciones de recompensa disponibles que puede personalizar:
  - Seguir la línea central. Recompense a su vehículo cuando conduzca de forma autónoma lo más cerca posible de la línea central de la pista.
  - Mantenerse dentro las líneas. Recompense a su vehículo cuando conduzca de forma autónoma, manteniendo las cuatro ruedas dentro de los límites de la pista.
  - Evitar la conducción en zigzag. Recompense a su vehículo por permanecer cerca de la línea central. Penalice a su vehículo si utiliza ángulos de dirección altos o se desvía del camino.

### Note

Si no quiere personalizar la función de recompensa, seleccione Siguiente.

2. (Opcional) Modifique el código de la función de recompensa.
  - Seleccione un ejemplo de función de recompensa y seleccione Explícame este código.
  - Para ver más información sobre cada sección del código, seleccione el signo + para que aparezca un cuadro de texto emergente con texto explicativo. Avance en el tutorial del código seleccionando Siguiente en cada ventana emergente. Para salir de un cuadro de texto emergente, seleccione la X de la esquina. Para abandonar el tutorial, seleccione Finalizar.

**Note**

Puede optar por no editar el código de la función de recompensa de muestra seleccionando Usar el código predeterminado.

- Si lo desea, edite el código de la función de recompensa de muestra seleccionando una función de recompensa de muestra y eligiendo Editar código de muestra. Edite el código y seleccione Validar para comprobar el código. Si el código no se puede validar o si quiere restablecerlo a su estado original, seleccione Restablecer.

3. Elija Siguiente.

## Paso 6: Elegir la duración y enviar su modelo a la tabla de clasificación

La duración del entrenamiento de su modelo afecta a su rendimiento. Cuando esté en la fase inicial del entrenamiento, debería empezar con un valor pequeño para este parámetro y luego entrenar progresivamente durante periodos de tiempo más prolongados.

En este paso del entrenamiento del modelo, el modelo entrenado se envía a una tabla de clasificación. Si no desea continuar, desmarque la casilla de verificación.

Para elegir la duración y enviar un modelo a la tabla de clasificación:

1. En la página Elegir duración, seleccione una hora en Elegir la duración del entrenamiento del modelo.
2. En el campo Descripción del modelo, introduzca una descripción útil para su modelo que le ayude a recordar las selecciones que ha realizado.

**Tip**

Se recomienda añadir información sobre el modelo, como las selecciones y modificaciones actuales de la función de recompensa y el algoritmo, así como sus hipótesis sobre el rendimiento del modelo.

3. Seleccione la casilla de verificación para que su modelo se envíe automáticamente a la tabla de clasificación de AWS DeepRacer Student una vez finalizado el entrenamiento. Si lo desea, puede optar por no introducir su modelo desmarcando la casilla de verificación.

 Tip

Le recomendamos que envíe su modelo a la tabla de clasificación. Al enviar su modelo, podrá compararlo con los demás y recibir información para que pueda mejorarlo.

4. Elija Entrene su modelo.
5. En la ventana emergente Iniciar el entrenamiento del modelo, seleccione Aceptar.
6. En la página de Configuración del entrenamiento, puede revisar el estado y la configuración del entrenamiento de su modelo. También puede ver un vídeo del entrenamiento de su modelo en la pista seleccionada cuando el Estado del entrenamiento esté En curso. Ver el vídeo puede ayudarle a obtener información valiosa que podrá utilizar para mejorar su modelo.

## Paso 7: Ver el rendimiento de su modelo en la tabla de clasificación

Una vez que haya entrenado su modelo y lo haya enviado a una tabla de clasificación, podrá ver su rendimiento.

Vea el rendimiento de su modelo

1. En el panel de navegación de la izquierda, vaya a Competir. Elija una temporada. En la página de Clasificación, aparecen su modelo y su rango en una sección. La página también incluye una sección de Clasificación con una lista de los modelos presentados, detalles de la carrera y una sección de Detalles de la carrera.
2. En la página que muestra la tabla de clasificación, en la sección de su perfil, seleccione Ver vídeo para ver un vídeo del rendimiento de su modelo.

## Paso 8: Usar Clonar para mejorar su modelo

Una vez que haya entrenado y, si lo desea, enviado su modelo a una tabla de clasificación, podrá clonarlo para mejorarlo. La clonación del modelo le ahorra pasos y hace que el entrenamiento sea más eficaz, ya que utiliza un modelo previamente entrenado como punto de partida para un modelo nuevo.

Para clonar y mejorar un modelo:

1. En el panel de navegación de la izquierda de AWS DeepRacer, vaya a Sus modelos.
2. En la página Sus modelos, seleccione un modelo y elija Clonar.
3. En el campo Asignar un nombre a su modelo, introduzca un nombre nuevo para el modelo clonado y seleccione Siguiente.
4. En la página Personalizar una función de recompensa, personalice la función de recompensa y luego seleccione Siguiente. Para obtener más información acerca de cómo personalizar una función de recompensa, consulte [Paso 5: Personalizar su función de recompensa](#).
5. En la página Elegir duración, introduzca una hora en el campo Elegir la duración del entrenamiento del modelo, introduzca una descripción en el campo Descripción del modelo y seleccione la casilla de verificación para enviar el modelo clonado a la tabla de clasificación.
6. Elija Entrene su modelo. Se ha iniciado su entrenamiento. Aparece la página Configuración del entrenamiento con información sobre el modelo clonado. También puede ver un vídeo del entrenamiento de su modelo en la pista seleccionada cuando el Estado del entrenamiento esté En curso.
7. Siga clonando y modificando sus modelos previamente entrenados para conseguir su mejor rendimiento en la clasificación.

## Paso 9: (opcional) Descargar un modelo

Después de entrenar un modelo y, si lo desea, de enviarlo a la tabla de clasificación, puede que desee descargarlo para usarlo en el futuro en un dispositivo físico de AWS DeepRacer. El modelo se guarda como un archivo `.tar.gz`.

Para descargar un archivo:

1. En el panel de navegación de la izquierda de AWS DeepRacer, vaya a Sus modelos.
2. En la página Sus modelos, seleccione un modelo y luego haga clic en Descargar.
3. Siga el estado de la descarga del modelo en su navegador. Cuando descargue su modelo, puede guardarlo en su disco duro local o en otro dispositivo de almacenamiento preferido.

Para obtener más información sobre cómo trabajar con dispositivos AWS DeepRacer, consulte [Opere su vehículo de AWS DeepRacer](#) en la guía AWS DeepRacer.

# Únase a una carrera de AWS DeepRacer Student

Después de entrenar y evaluar con éxito su modelo en simulación, compare el rendimiento de su modelo con el de otros corredores participando en una carrera. Las carreras son una forma estupenda de recibir comentarios sobre su modelo, ganar premios y galardones, conocer virtualmente a otros miembros de la comunidad de AWS DeepRacer Student, enterarse de oportunidades para aprender, mejorar sus habilidades y divertirse. Existen dos tipos de carreras de estudiantes: una carrera de la Student League y una carrera comunitaria.

Una carrera de la Student League es una competición virtual mensual en la que pueden participar todos los estudiantes. Una carrera comunitaria de estudiantes es una carrera privada creada por un educador o un organizador de eventos en la consola de AWS y a la que los estudiantes solo pueden unirse por invitación.

Esta sección explica cómo participar en una carrera de la Student League y en una carrera comunitaria de AWS DeepRacer Student.

## Únase a una carrera de la Student League

En esta sección, aprenderá a enviar su modelo a una carrera de la Student League de AWS DeepRacer Student. Puede unirse a una carrera todos los meses enviando un modelo entrenado directamente a la clasificación de estudiantes. Para obtener más información acerca de los modelos de entrenamiento, consulte [Entrenamiento de un modelo de aprendizaje por refuerzo en AWS DeepRacer Student](#).

Para unirse a una carrera de la Student League:

1. En el panel de navegación izquierdo de AWS DeepRacer Student, expanda Competir y seleccione Student League.
2. La página muestra los detalles de la carrera y la tabla de clasificación. En la sección Carreras abiertas, seleccione Participar en la carrera para enviar un modelo o, si aún no lo ha creado, seleccione Crear modelo. Para obtener más información, consulte [Entrenamiento de un modelo de aprendizaje por refuerzo en AWS DeepRacer Student](#).
3. En la sección Elegir un modelo para correr, use la lista desplegable para elegir un modelo en el campo Elija un modelo.
4. Seleccione Unirse a la carrera para enviar su modelo.

5. Si su modelo se evalúa con éxito en función de los criterios de competición, consulte la tabla de clasificación del evento para ver en qué posición se encuentra su modelo con respecto a los demás participantes.
6. Si lo desea, seleccione Ver para ver un vídeo del rendimiento de su vehículo.
7. Seleccione Participar de nuevo para participar con otro modelo. El cupo de inscripciones para cada carrera de la Student League de AWS DeepRacer es de 50.

## Únase a una carrera comunitaria de estudiantes

En esta sección, aprenderá a enviar su modelo a una carrera comunitaria de estudiantes de AWS DeepRacer Student. Puede unirse a una carrera comunitaria de estudiantes si recibe un enlace de invitación de su educador u organizador del evento por correo electrónico.

Para unirse a una carrera comunitaria de estudiantes:

1. Vaya al enlace de invitación e inicie sesión en su cuenta de AWS DeepRacer Student.
2. Una vez que haya iniciado sesión, pulse el botón Participar en la carrera.
3. En la lista desplegable Elija un modelo para correr, seleccione su modelo para usarlo en la carrera comunitaria.
4. Seleccione Participar en la carrera para enviar su modelo.
5. Si su modelo se evalúa con éxito en función de los criterios de competición, compruebe la tabla de clasificación del evento para ver cómo su modelo se clasifica frente a otros participantes.
6. Si lo desea, seleccione Ver para ver un vídeo del rendimiento de su vehículo.
7. Seleccione Participar de nuevo para participar con otro modelo.

# Personalizar una función de recompensa

Crear una función de recompensa es como diseñar un plan de incentivos. Los parámetros son valores que se pueden utilizar para desarrollar su plan de incentivos.

Según la estrategia de incentivos que se diseñe, el comportamiento del vehículo será diferente. Para incentivar al vehículo a conducir más rápido, intente conceder valores negativos cuando el vehículo tarde demasiado en terminar una vuelta o se salga de la pista. Para evitar patrones de conducción en zigzag, pruebe a definir un límite de ángulo de dirección y premie al coche por conducir menos agresivamente en los tramos rectos de la pista.

Puede utilizar los waypoints, que son marcadores numerados situados a lo largo de la línea central y los bordes exteriores e interiores de la pista, para ayudarle a asociar determinados comportamientos de conducción con características específicas de una pista, como rectas y curvas.

Crear una función de recompensa eficaz es un proceso creativo y repetitivo. Pruebe diferentes estrategias, combine parámetros y, lo más importante, ¡diviértase!

## Temas

- [Editar el código Python para personalizar su función de recompensa](#)
- [Parámetros de entrada de la función de recompensa de AWS DeepRacer](#)

## Editar el código Python para personalizar su función de recompensa

En AWS DeepRacer Student, puede editar ejemplos de funciones de recompensa para diseñar una estrategia de carreras personalizada para su modelo.

Para personalizar su función de recompensa:

1. En la página Paso 5: Personalizar la función de recompensa de la página de AWS DeepRacer Student Crear modelo, seleccione un ejemplo de función de recompensa.
2. Utilice el editor de código situado debajo del selector de funciones de recompensa de ejemplo para personalizar los parámetros de entrada de la función de recompensa mediante el código Python.
3. Seleccione Validar para comprobar si su código funcionará o no. Como alternativa, seleccione Restablecer para volver a empezar.

#### 4. Cuando termine de realizar cambios, seleccione Siguiente.

Utilice [Parámetros de entrada de la función de recompensa de AWS DeepRacer](#) para obtener información sobre cada parámetro. Consulte cómo se utilizan los distintos parámetros en los ejemplos de funciones de recompensa.

## Parámetros de entrada de la función de recompensa de AWS DeepRacer

La función de recompensa de AWS DeepRacer coge un objeto del diccionario aprobado como variable, `params`, como entrada.

```
def reward_function(params) :  
  
    reward = ...  
  
    return float(reward)
```

El objeto del diccionario `params` contiene los siguientes pares de clave-valor:

```
{  
    "all_wheels_on_track": Boolean,          # flag to indicate if the agent is on the  
    track  
    "x": float,                             # agent's x-coordinate in meters  
    "y": float,                             # agent's y-coordinate in meters  
    "closest_objects": [int, int],          # zero-based indices of the two closest  
    objects to the agent's current position of (x, y).  
    "closest_waypoints": [int, int],        # indices of the two nearest waypoints.  
    "distance_from_center": float,          # distance in meters from the track center  
    "is_crashed": Boolean,                  # Boolean flag to indicate whether the agent  
    has crashed.  
    "is_left_of_center": Boolean,           # Flag to indicate if the agent is on the  
    left side to the track center or not.  
    "is_offtrack": Boolean,                 # Boolean flag to indicate whether the agent  
    has gone off track.  
    "is_reversed": Boolean,                 # flag to indicate if the agent is driving  
    clockwise (True) or counter clockwise (False).  
    "heading": float,                       # agent's yaw in degrees  
    "objects_distance": [float, ],          # list of the objects' distances in meters  
    between 0 and track_length in relation to the starting line.
```

```
"objects_heading": [float, ], # list of the objects' headings in degrees
between -180 and 180.
"objects_left_of_center": [Boolean, ], # list of Boolean flags indicating whether
elements' objects are left of the center (True) or not (False).
"objects_location": [(float, float),], # list of object locations [(x,y), ...].
"objects_speed": [float, ], # list of the objects' speeds in meters per
second.
"progress": float, # percentage of track completed
"speed": float, # agent's speed in meters per second (m/s)
"steering_angle": float, # agent's steering angle in degrees
"steps": int, # number steps completed
"track_length": float, # track length in meters.
"track_width": float, # width of the track
"waypoints": [(float, float), ] # list of (x,y) as milestones along the
track center
}
```

Utilice la siguiente referencia para comprender mejor los parámetros de entrada de AWS DeepRacer.

## all\_wheels\_on\_track

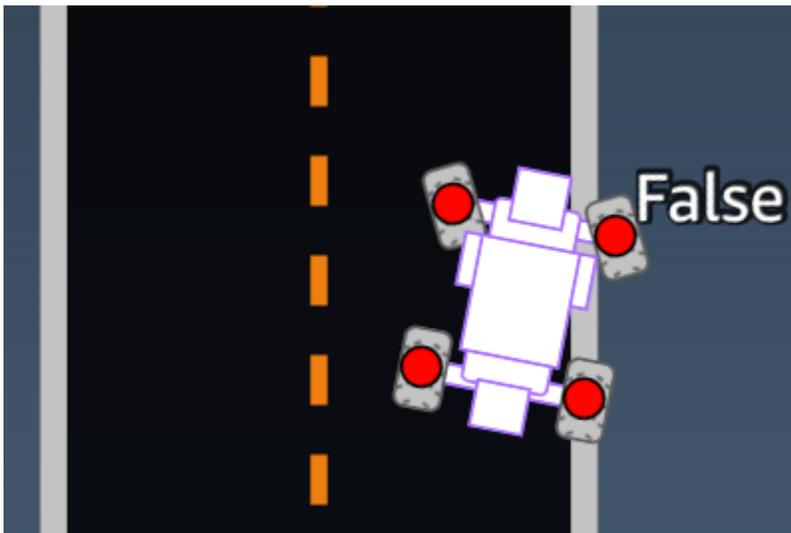
Tipo: Boolean

Rango: (True:False)

Una bandera Boolean para indicar si el agente está en la pista o fuera de ella. El agente está fuera de la pista (False) si cualquiera de sus ruedas está fuera de los límites de la pista. Está en la pista (True) si todas las ruedas están dentro de los bordes interiores y exteriores de la pista. La siguiente ilustración muestra un agente que está dentro de la pista.



La siguiente ilustración muestra un agente que no está en la pista debido a que dos ruedas están fuera de los bordes de la pista.



Ejemplo: una función de recompensa utilizando el parámetro `all_wheels_on_track`.

```
def reward_function(params):  
    #####  
    ...  
    Example of using all_wheels_on_track and speed  
    ...
```

```
# Read input variables
all_wheels_on_track = params['all_wheels_on_track']
speed = params['speed']

# Set the speed threshold based your action space
SPEED_THRESHOLD = 1.0

if not all_wheels_on_track:
    # Penalize if the car goes off track
    reward = 1e-3
elif speed < SPEED_THRESHOLD:
    # Penalize if the car goes too slow
    reward = 0.5
else:
    # High reward if the car stays on track and goes fast
    reward = 1.0

return float(reward)
```

## closest\_waypoints

Tipo: [int, int]

Rango: [(0:Max-1), (1:Max-1)]

Los índices de base cero de los dos waypoint más cercanos a la posición actual (x, y) del agente. La distancia se mide por la distancia euclidiana desde el centro del agente. El primer elemento se refiere al punto de ruta más cercano detrás del agente y el segundo elemento hace referencia al punto de ruta más cercano delante del agente. Max es la longitud de la lista de puntos de ruta. En la ilustración que se muestra en [waypoints](#), los `closest_waypoints` son [16, 17].

En el siguiente ejemplo, se muestra cómo utilizar la función de recompensa `waypoints` y `closest_waypoints`, así como `heading` para calcular las recompensas inmediatas.

AWS DeepRacer es compatible con las siguientes bibliotecas de Python: `math`, `random`, `numpy`, `scipy` y `shapely`. Para utilizarlas, añade una declaración de importación, `import supported library`, antes de la definición de la función, `def reward_function(params)`.

Ejemplo: una función de recompensa utilizando el parámetro `closest_waypoints`.

```
# Place import statement outside of function (supported libraries: math, random, numpy,
  scipy, and shapely)
# Example imports of available libraries
#
# import math
# import random
# import numpy
# import scipy
# import shapely

import math

def reward_function(params):
    #####
    '''
    Example of using waypoints and heading to make the car point in the right direction
    '''

    # Read input variables
    waypoints = params['waypoints']
    closest_waypoints = params['closest_waypoints']
    heading = params['heading']

    # Initialize the reward with typical value
    reward = 1.0

    # Calculate the direction of the centerline based on the closest waypoints
    next_point = waypoints[closest_waypoints[1]]
    prev_point = waypoints[closest_waypoints[0]]

    # Calculate the direction in radius, arctan2(dy, dx), the result is (-pi, pi) in
    radians
    track_direction = math.atan2(next_point[1] - prev_point[1], next_point[0] -
    prev_point[0])
    # Convert to degree
    track_direction = math.degrees(track_direction)

    # Calculate the difference between the track direction and the heading direction of
    the car
    direction_diff = abs(track_direction - heading)
    if direction_diff > 180:
        direction_diff = 360 - direction_diff
```

```
# Penalize the reward if the difference is too large
DIRECTION_THRESHOLD = 10.0
if direction_diff > DIRECTION_THRESHOLD:
    reward *= 0.5

return float(reward)
```

## closest\_objects

Tipo: [int, int]

Rango: [(0:len(object\_locations)-1), (0:len(object\_locations)-1)]

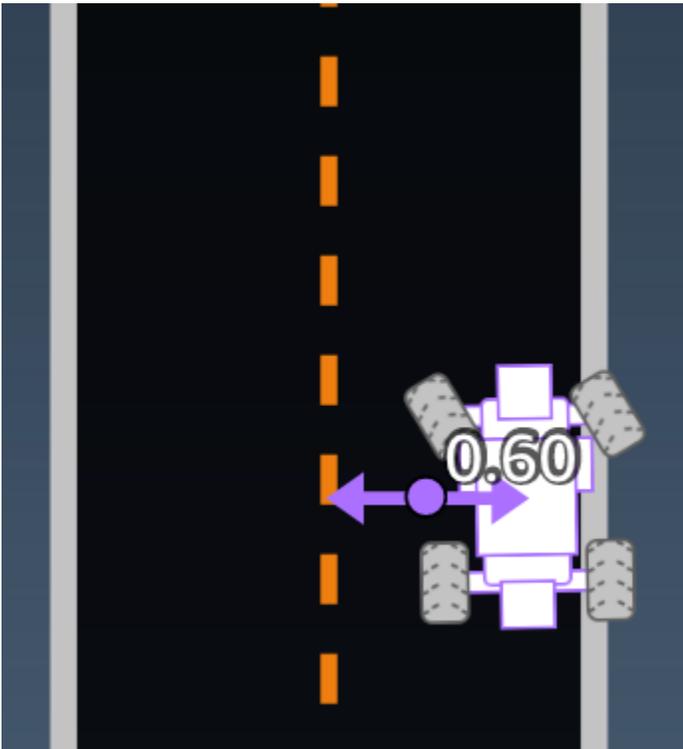
Índices basados en cero de los dos objetos más cercanos a la posición actual (x,y) del agente. El primer índice hace referencia al objeto más cercano situado detrás del agente, mientras que el segundo índice hace referencia al objeto más cercano situado delante. Si solo hay un objeto, ambos índices serán 0.

## distance\_from\_center

Tipo: float

Rango: 0:~track\_width/2

Desplazamiento, en metros, entre el centro del agente y el centro de la pista. El desplazamiento máximo observable se produce cuando cualquiera de las ruedas del agente están fuera del límite de la pista y, en función de la anchura de la pista, puede ser ligeramente más pequeña o más grande que la mitad de track\_width.



Ejemplo: una función de recompensa utilizando el parámetro *distance\_from\_center*.

```
def reward_function(params):  
    #####  
    ...  
    Example of using distance from the center  
    ...  
  
    # Read input variable  
    track_width = params['track_width']  
    distance_from_center = params['distance_from_center']  
  
    # Penalize if the car is too far away from the center  
    marker_1 = 0.1 * track_width  
    marker_2 = 0.5 * track_width  
  
    if distance_from_center <= marker_1:  
        reward = 1.0  
    elif distance_from_center <= marker_2:  
        reward = 0.5  
    else:  
        reward = 1e-3 # likely crashed/ close to off track  
  
    return float(reward)
```

## heading

Tipo: float

Rango: -180:+180

La dirección del rumbo, en grados, del agente con respecto al eje x del sistema de coordenadas.



Ejemplo: una función de recompensa utilizando el parámetro *heading*.

Para obtener más información, consulte [closest\\_waypoints](#).

## is\_crashed

Tipo: Boolean

Rango: (True:False)

Una bandera Boolean que indica si el agente se chocó contra otro objeto (True) o no (False) como estado de terminación.

## is\_left\_of\_center

Tipo: Boolean

Rango: [True : False]

Una bandera Boolean que indica si el agente está a la izquierda del centro de la pista (True) o no está a la izquierda del centro de la pista (False).

## is\_offtrack

Tipo: Boolean

Rango: (True:False)

Una bandera Boolean que indica si las cuatro ruedas del agente se han desplazado fuera de los bordes interiores o exteriores de la pista (True) o no (False).

## is\_reversed

Tipo: Boolean

Rango: [True:False]

Una bandera Boolean que indica si el agente conduce en el sentido de las agujas del reloj (True) o en el sentido contrario a las agujas del reloj (False).

Se utiliza cuando se habilita el cambio de dirección para cada episodio.

## objects\_distance

Tipo: [float, ... ]

Rango: [(0:track\_length), ... ]

Una lista de distancias entre objetos del entorno en relación con la línea de salida. El elemento  $i^{\text{th}}$  mide la distancia en metros entre el objeto  $i^{\text{th}}$  y la línea de salida a lo largo de la línea central de la pista.

### Note

$\text{abs} | (\text{var1}) - (\text{var2}) |$  = a qué distancia está el coche de un objeto, WHEN  $\text{var1} = [\text{"objects\_distance"}][\text{index}]$  y  $\text{var2} = \text{params}[\text{"progress"}]*\text{params}[\text{"track\_length"}]$

Para obtener un índice del objeto más cercano delante del vehículo y del objeto más cercano detrás del vehículo, utilice el parámetro `closest_objects`.

## objects\_heading

Tipo: [float, ... ]

Rango: [(-180:180), ... ]

Lista de los encabezados de los objetos en grados. El elemento  $i^{\text{th}}$  mide el encabezado del objeto  $i^{\text{th}}$ . Los encabezados de los objetos estacionarios son 0. Para un vehículo robot, el valor del elemento correspondiente es el ángulo de dirección del vehículo robot.

## objects\_left\_of\_center

Tipo: [Boolean, ... ]

Rango: [True|False, ... ]

Lista de banderas Boolean. El valor del elemento  $i^{\text{th}}$  indica si el objeto  $i^{\text{th}}$  está en el lado izquierdo (True) o derecho (False) del centro de la pista.

## objects\_location

Tipo: [(x,y), ...]

Rango: [(0:N,0:N), ...]

Este parámetro almacena todas las ubicaciones de los objetos. Cada ubicación es una tupla de ([x](#), [y](#)).

El tamaño de la lista es igual al número de objetos en la pista. Los objetos de la lista incluyen tanto obstáculos estacionarios como vehículos robot en movimiento.

## objects\_speed

Tipo: [float, ... ]

Rango: [(0:12.0), ... ]

Lista de velocidades (metros por segundo) para los objetos en la pista. En objetos estacionarios, las velocidades son 0. En un vehículo robot, el valor es la velocidad que se establece en el entrenamiento.

## progress

Tipo: float

Rango: 0:100

Porcentaje de pista completado.

Ejemplo: una función de recompensa utilizando el parámetro *progress*.

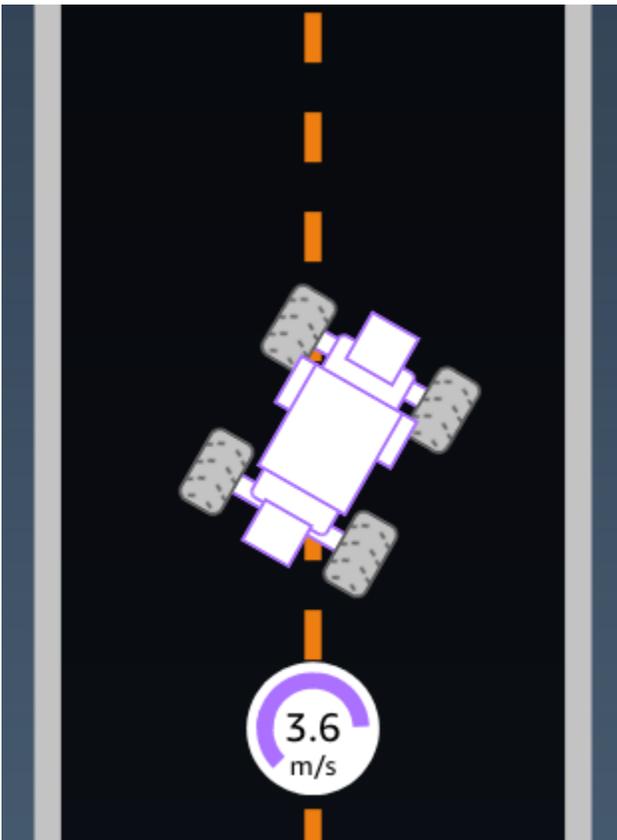
Para obtener más información, consulte [pasos](#).

## speed

Tipo: float

Rango: 0.0:5.0

La velocidad observada del agente, en metros por segundo (m/s).



Ejemplo: una función de recompensa utilizando el parámetro *speed*.

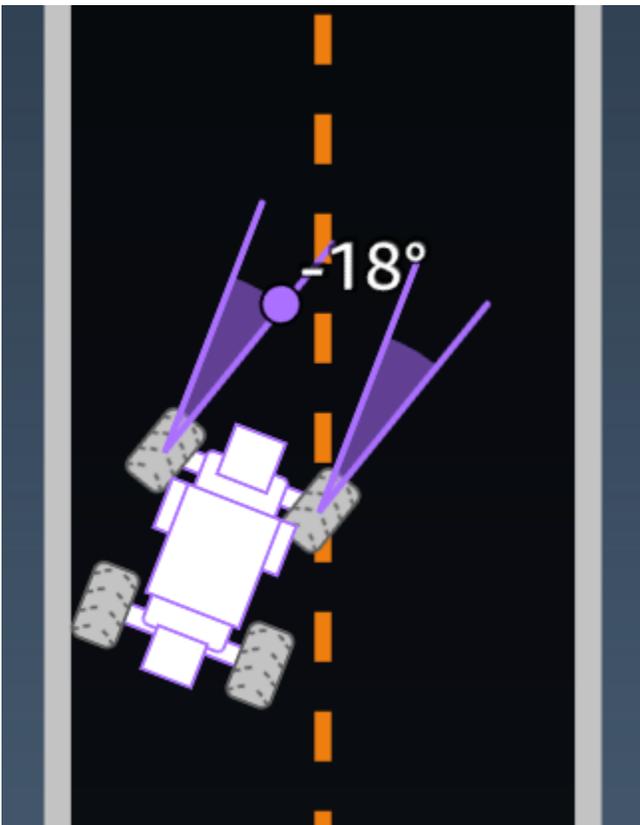
Para obtener más información, consulte la sección [all\\_wheels\\_on\\_track](#).

## steering\_angle

Tipo: float

Rango: -30:30

Ángulo de dirección, en grados, de las ruedas delanteras desde la línea central del agente. El signo negativo (-) significa maniobrar hacia la derecha y el signo positivo (+) significa maniobrar hacia la izquierda. La línea central del agente no es necesariamente paralela a la línea central de la pista, como se muestra en la siguiente ilustración.



Ejemplo: una función de recompensa utilizando el parámetro *steering\_angle*.

```
def reward_function(params):  
    '''  
    Example of using steering angle  
    '''  
  
    # Read input variable
```

```

    abs_steering = abs(params['steering_angle']) # We don't care whether it is left or
right steering

    # Initialize the reward with typical value
    reward = 1.0

    # Penalize if car steer too much to prevent zigzag
    ABS_STEERING_THRESHOLD = 20.0
    if abs_steering > ABS_STEERING_THRESHOLD:
        reward *= 0.8

    return float(reward)

```

## pasos

Tipo: int

Rango: 0:N<sub>step</sub>

Número de pasos finalizados. Un paso corresponde a una secuencia de observación-acción completada por el agente utilizando la política actual.

Ejemplo: una función de recompensa utilizando el parámetro *steps*.

```

def reward_function(params):
    #####
    '''
    Example of using steps and progress
    '''

    # Read input variable
    steps = params['steps']
    progress = params['progress']

    # Total num of steps we want the car to finish the lap, it will vary depends on the
track length
    TOTAL_NUM_STEPS = 300

    # Initialize the reward with typical value
    reward = 1.0

    # Give additional reward if the car pass every 100 steps faster than expected
    if (steps % 100) == 0 and progress > (steps / TOTAL_NUM_STEPS) * 100 :

```

```
reward += 10.0  
  
return float(reward)
```

## track\_length

Tipo: float

Rango:  $[0:L_{\max}]$

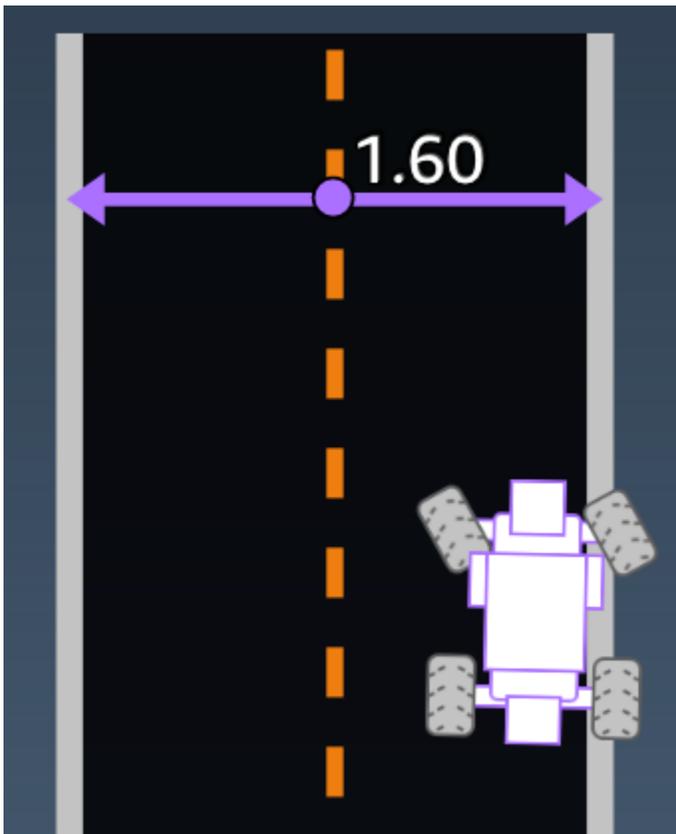
La longitud de la pista en metros.  $L_{\max}$  is track-dependent.

## track\_width

Tipo: float

Rango:  $0:D_{\text{track}}$

Ancho de la pista en metros.



Ejemplo: una función de recompensa utilizando el parámetro *track\_width*.

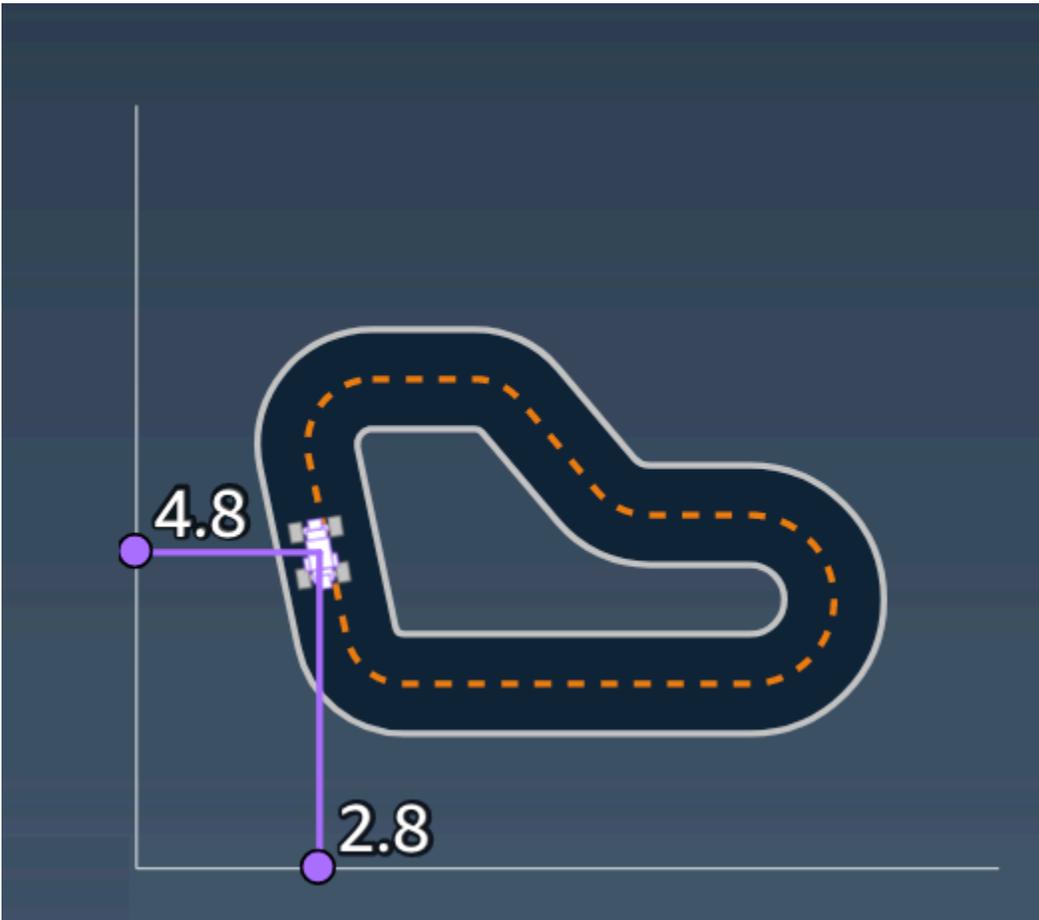
```
def reward_function(params):  
    #####  
    '''  
    Example of using track width  
    '''  
  
    # Read input variable  
    track_width = params['track_width']  
    distance_from_center = params['distance_from_center']  
  
    # Calculate the distance from each border  
    distance_from_border = 0.5 * track_width - distance_from_center  
  
    # Reward higher if the car stays inside the track borders  
    if distance_from_border >= 0.05:  
        reward = 1.0  
    else:  
        reward = 1e-3 # Low reward if too close to the border or goes off the track  
  
    return float(reward)
```

**X, y**

Tipo: float

Rango: 0:N

Localización, en metros, del centro del agente a lo largo de los ejes x e y del entorno simulado que contiene la pista. El origen se encuentra en el ángulo inferior izquierdo del entorno simulado.

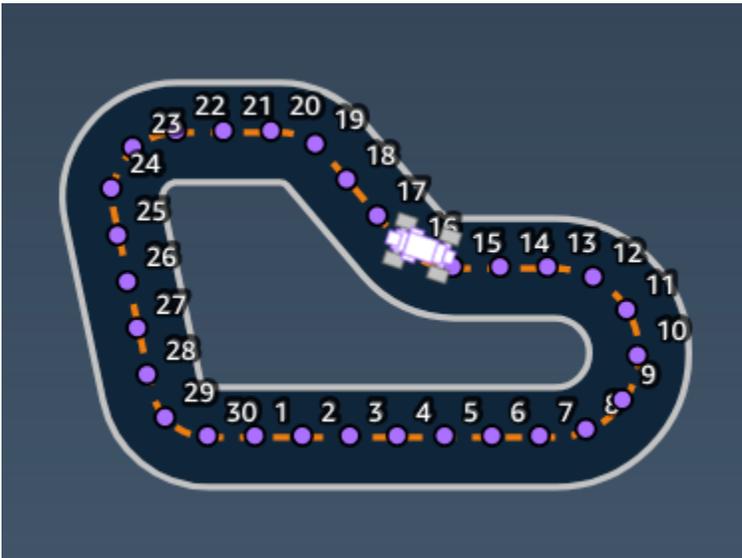


## waypoints

Tipo: list de [float, float]

Rango:  $[[x_{w,0}, y_{w,0}] \dots [x_{w,Max-1}, y_{w,Max-1}]]$

Una lista ordenada de hitos Max dependientes de la pista a lo largo del centro de la pista. Cada hito se describe mediante una coordenada de  $(x_{w,i}, y_{w,i})$ . Para una pista en bucle, el primer y el último señalador son los mismos. Para una pista recta u otra pista no en bucle, el primer y el último señalador son distintos.



Ejemplo: una función de recompensa utilizando el parámetro *waypoints*.

Para obtener más información, consulte [closest\\_waypoints](#).

# Seguridad en AWS DeepRacer Student

La seguridad en la nube de AWS es la mayor prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y usted. El [modelo de responsabilidad compartida](#) la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los [Programas de conformidad de AWS](#) . Para obtener información sobre los programas de conformidad que se aplican a AWS DeepRacer Student, consulte [Servicios de AWS en el ámbito del programa de conformidad](#).
- Seguridad en la nube: su responsabilidad viene determinada por el servicio de AWS que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando utiliza AWS DeepRacer Student. Muestra cómo configurar AWS DeepRacer Student para satisfacer sus objetivos de seguridad y conformidad. También puede aprender a utilizar otros servicios de AWS que le ayuden a monitorear y proteger sus recursos de AWS DeepRacer Student.

## Contenido

- [Protección de los datos en AWS DeepRacer Student](#)
- [Administración de identidades y accesos para AWS DeepRacer Student](#)
- [Validación de conformidad para AWS DeepRacer Student](#)
- [La resiliencia en AWS DeepRacerStudent](#)
- [Seguridad de la infraestructura en AWS DeepRacer Student](#)

## Protección de los datos en AWS DeepRacer Student

En las siguientes secciones se explica qué datos captura AWS DeepRacer Student y dónde utiliza el cifrado de datos para proteger sus datos.

Cuando crea una cuenta de AWS DeepRacer Student, también crea una cuenta de AWS jugador. Los recursos creados en su cuenta de AWS DeepRacer Student se almacenan en su cuenta de AWS jugador. Para obtener más información sobre las cuentas de AWS jugador, consulte [¿Qué son las cuentas de AWS jugador?](#) en la Guía de usuario de AWS DeepRacer Student.

## Temas

- [Datos almacenados en el portal AWS DeepRacer Student](#)
- [El cifrado en reposo en el portal AWS DeepRacer Student](#)
- [Cifrado en tránsito en el portal AWS DeepRacer Student](#)

## Datos almacenados en el portal AWS DeepRacer Student

Para utilizar el portal AWS DeepRacer Student, los datos necesarios se almacenan en su cuenta de AWS jugador. Los datos recopilados en el portal AWS DeepRacer Student no se utilizan para ayudar a mejorar el servicio.

Datos capturados en AWS DeepRacer Student.

El siguiente es un resumen de los datos creados en AWS DeepRacer Student y almacenados en su cuenta de AWS jugador.

- Su dirección de correo electrónico y contraseña utilizadas para registrar su cuenta
- Su nombre de corredor
- Su posición en la clasificación de la Student League
- Sus modelos entrenados
- Código de la función de recompensa

## El cifrado en reposo en el portal AWS DeepRacer Student

Los datos capturados por el portal AWS DeepRacer Student están cifrados de forma predeterminada.

Las cuentas de AWS jugador utilizan Amazon Cognito para cifrar y almacenar el correo electrónico y la contraseña utilizados para iniciar sesión en AWS DeepRacer Student. Para obtener más información, consulte la [Protección de datos en Amazon Cognito](#).

Todos los demás datos capturados en AWS DeepRacer Student se cifran en reposo en la nube mediante claves propias de AWS a través de AWS Key Management Service con AES-GCM y

utilizando claves de 256 bits. Estos datos se almacenan y cifran en Amazon Simple Storage Service (S3) y Amazon DynamoDB.

## Cifrado en tránsito en el portal AWS DeepRacer Student

Sus direcciones de correo electrónico registradas y autorizadas se cifran con el cifrado del cliente. Todos los demás [datos capturados en AWS DeepRacer Student](#) se copian de su cuenta y se procesan en un sistema interno de AWS. De forma predeterminada, AWS DeepRacer Student utiliza conexiones seguras a través de HTTPS para cifrar los datos en tránsito.

## Administración de identidades y accesos para AWS DeepRacer Student

AWS Identity and Access Management (IAM) es un servicio de AWS que ayuda a los administradores a controlar de forma segura el acceso a los recursos de AWS. Los administradores de IAM controlan quién puede estar AWSautenticado (ha iniciado sesión) y autorizado (tiene permisos) para utilizar recursos de AWS DeepRacer Student no se integra directamente con IAM para controlar el acceso de los usuarios a los recursos de AWS. En su lugar, AWS DeepRacer Student utiliza una API proxy autenticada gestionada por AWS DeepRacer para proteger los recursos de los usuarios.

## Validación de conformidad para AWS DeepRacer Student

Audidores externos evalúan la seguridad y la conformidad de AWS DeepRacer Student como parte de varios programas de conformidad de AWS.

Para obtener una lista de servicios de AWS en el ámbito de programas de conformidad específicos, consulte [Servicios de AWS en el ámbito del programa de conformidad](#). Para obtener información general, consulte [Programas de conformidad de AWS](#).

Puede descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#).

AWS proporciona los siguientes recursos para ayudar con la conformidad:

- [Guías de inicio rápido de seguridad y conformidad](#): estas guías de implementación tratan consideraciones sobre arquitectura y ofrecen pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.

- [Recursos de conformidad de AWS](#): este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- [Evaluación de recursos con reglas](#) en la Guía para desarrolladores de AWS Config; AWS Config evalúa en qué medida las configuraciones de sus recursos cumplen las prácticas internas, las directrices del sector y las normativas.
- [AWS Security Hub](#) – Este servicio de AWS proporciona una vista integral de su estado de seguridad en AWS que ayuda a verificar la conformidad con los estándares y las prácticas recomendadas del sector de seguridad.

## La resiliencia en AWS DeepRacerStudent

La infraestructura global de AWS está conformada por regiones y zonas de disponibilidad de AWS. Las regiones proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre las regiones y zonas de disponibilidad de AWS, consulte la [Infraestructura global de AWS](#).

## Seguridad de la infraestructura en AWS DeepRacer Student

Al tratarse de un servicio administrado, AWS DeepRacer Student está protegido por los procedimientos de seguridad de red globales de AWS que se describen en el documento técnico [Amazon Web Services: Información general sobre los procesos de seguridad](#).

# Solución de problemas comunes AWS DeepRacer de los estudiantes

## Temas

- [¿Por qué se cerró sesión automáticamente en mi cuenta de AWS DeepRacer estudiante?](#)
- [¿Cómo puedo dejar de participar en el programa de becas de IA y ML de AWS?](#)
- [No puedo eliminar mi cuenta AWS DeepRacer de estudiante](#)
- [No encuentro el nombre de mi escuela en la lista desplegable](#)
- [No puedo seguir entrenando a mi modelo](#)
- [Aparece el mensaje de error “Hay una cuenta registrada con este correo electrónico”](#)
- [Me he registrado con una cuenta de Gmail y no encuentro mi código de verificación](#)

## ¿Por qué se cerró sesión automáticamente en mi cuenta de AWS DeepRacer estudiante?

De conformidad con AWS la política de seguridad, se cierra automáticamente la sesión de tu cuenta de AWS DeepRacer estudiante después de 30 días.

- Para seguir utilizando el servicio, dirígete a la [página de inicio de sesión para AWS DeepRacer estudiantes](#) y utiliza tus credenciales para volver a iniciar sesión.

## ¿Cómo puedo dejar de participar en el programa de becas de IA y ML de AWS?

El programa de becas de IA y ML de AWS es opcional y está destinado a estudiantes de 16 años o más que carecen de recursos necesarios o de representación suficiente. Cuando se inscribe en AWS DeepRacer Student, de forma predeterminada no está inscrito en el programa de becas AWS AI & ML.

Para participar, primero debe inscribirse marcando la casilla de la sección ¿Desea que se le considere para el programa de becas de IA y aprendizaje automático de AWS? a medida que se inscribe en AWS DeepRacer Student or later desde la página Tu perfil, a la que se puede acceder desde el panel de navegación izquierdo del sitio.

- Si desea ingresar en el programa, solo tendrá acceso al proceso de solicitud. También puede optar por no presentar la solicitud.

## No puedo eliminar mi cuenta AWS DeepRacer de estudiante

Si no puedes eliminar tu cuenta de AWS DeepRacer estudiante, comprueba si has creado un evento AWS DeepRacer multiusuario. AWS Las cuentas de jugador son una solución de identidad gestionada creada AWS por AWS DeepRacer varios usuarios y AWS DeepRacer estudiantes. Tu cuenta de AWS jugador contiene todos los recursos creados en cada uno de estos AWS servicios.

- Para garantizar que los participantes de los eventos que crees no se queden sin experiencia, no puedes eliminar tu cuenta de AWS DeepRacer estudiante si contiene recursos para un evento AWS DeepRacer multiusuario.

## No encuentro el nombre de mi escuela en la lista desplegable

Es posible que no encuentre todas las escuelas en la lista desplegable, especialmente las secundarias.

- Si su centro educativo no está en la lista desplegable, seleccione Otros e introduzca el nombre del centro educativo.

## No puedo seguir entrenando a mi modelo

Es posible que haya superado el límite mensual de horas de entrenamiento del modelo.

- Vaya a la página Inicio para comprobar las horas de entrenamiento utilizadas en la sección de Horas restantes de entrenamiento del modelo. Si ha superado las horas de entrenamiento del modelo, espere a que se restablezcan las horas para volver a entrenarlo.

## Aparece el mensaje de error “Hay una cuenta registrada con este correo electrónico”

Recibirás este mensaje de error cuando introduzcas un código de confirmación en la página de registro de la cuenta de AWS jugador y ya hayas utilizado la misma dirección de correo electrónico

para crear una cuenta de AWS jugador a través de la AWS DeepRacer función multiusuario. También recibes este error si anteriormente utilizaste la misma dirección de correo electrónico para crear una cuenta de AWS DeepRacer estudiante.

- Inicia sesión en la [página de inicio de sesión para AWS DeepRacer estudiantes](#) con las credenciales que creaste anteriormente o solicita un restablecimiento de contraseña seleccionando ¿Has olvidado tu contraseña? en el campo Contraseña.

## Me he registrado con una cuenta de Gmail y no encuentro mi código de verificación

Si has creado una cuenta de AWS jugador con una cuenta de Gmail y no encuentras el mensaje con el código de verificación, es posible que se haya enviado a la carpeta equivocada.

- Inicie sesión en su cuenta de Gmail y consulte la carpeta de Promociones y busque el mensaje titulado “Su código de verificación de perfil de jugador de AWS”.

# Cuotas para AWS DeepRacer Student

Cada estudiante que participe en AWS DeepRacer Student recibe 10 horas mensuales gratuitas de recursos informáticos de entrenamiento de modelos y 5GB de almacenamiento.

# Eliminar tu cuenta AWS DeepRacer de estudiante

El portal AWS DeepRacer para estudiantes almacena la siguiente información en tu cuenta de AWS jugador:

- Dirección de correo electrónico
- Su contraseña
- Su nombre de corredor
- Su posición en la tabla de clasificación de la Student League

Para obtener más información sobre los datos recopilados, consulta [Protección de los datos en AWS DeepRacer Student](#) la Guía del usuario para AWS DeepRacer estudiantes.

Si desea eliminar esta información AWS de los servidores, utilice el siguiente procedimiento para eliminar su cuenta del portal AWS DeepRacer estudiantil. Al eliminar su cuenta de AWS DeepRacer estudiante, también se eliminará su cuenta de AWS jugador y todos los recursos asociados.

Para obtener más información sobre las cuentas de AWS jugador, consulta [¿Qué son las cuentas de AWS jugador?](#)

## Note

Si has creado un evento AWS DeepRacer multiusuario, no puedes eliminar tu cuenta de AWS jugador. Para obtener más información, consulta [No puedo eliminar mi cuenta AWS DeepRacer de estudiante](#) la Guía del usuario AWS DeepRacer para estudiantes.

Para eliminar tu cuenta de jugador AWS DeepRacer estudiantil

## Important

Eliminar tu cuenta de AWS DeepRacer estudiante es una acción que no se puede deshacer. Cuando eliminas a tu AWS DeepRacer estudiante, también eliminas tu cuenta de AWS jugador y todos los recursos asociados.

Cuando eliminas tu cuenta de AWS DeepRacer estudiante, los recursos de tu cuenta de AWS jugador se eliminan de nuestros servidores en el plazo de un año.

1. Abre la página de inicio AWS DeepRacer para estudiantes: <https://student.deepracer.com/signIn>.
2. Si se le solicita, inicie sesión en su cuenta de AWS DeepRacer estudiante.
3. Elija Su cuenta.
4. En la página Su cuenta, seleccione Eliminar su cuenta.
5. En Para confirmar la eliminación, escriba Eliminar en el campo, escriba **Delete**.
6. Elija Eliminar.

Cuando su cuenta se haya eliminado correctamente, aparecerá el mensaje La cuenta se ha eliminado correctamente y volverá a la página de inicio de sesión del AWS DeepRacer estudiante.

Si también deseas eliminar tu AWS cuenta, sigue los pasos que se describen en [Cerrar tu AWS cuenta](#).

Sabemos que los clientes se preocupan mucho por la privacidad y la seguridad de los datos, por lo que implementamos controles técnicos y físicos responsables y sofisticados diseñados para evitar el acceso no autorizado al contenido del cliente o su divulgación. Mantener la confianza de los clientes es un compromiso continuo. Puedes obtener más información sobre AWS los compromisos de privacidad de los datos en nuestra página [de Aviso de privacidad](#).

# Historial de revisión de la Guía del usuario de AWS DeepRacer Student

En la siguiente tabla se describen las versiones de la documentación del AWS DeepRacer Student.

Cambio	Descripción	Fecha
<a href="#">Actualizaciones para el programa de 2024</a>	Actualizaciones de temas relacionados con el programa de becas de IA y ML de AWS de 2024 y la Student League de AWS DeepRacer. Se agregó información sobre el nuevo curso, disponible para todos los estudiantes que soliciten una beca de IA y ML de AWS.	1 de noviembre de 2023
<a href="#">Actualizaciones de la Student League de AWS DeepRacer de 2023.</a>	Actualizaciones de varios temas relacionados con la Student League de AWS DeepRacer de 2023. Para obtener más información, consulte los <a href="#">Términos y condiciones</a> .	1 de marzo de 2023
<a href="#">Versión inicial</a>	Versión inicial de la Guía de usuario de AWS DeepRacer Student, que incluye soporte para el programa de becas de IA y ML de AWS y la Student League de AWS DeepRacer.	1 de diciembre de 2021

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la versión original de inglés, prevalecerá la versión en inglés.