aws

Amazon Mechanical Turk

API Version 2017-01-17

Amazon Mechanical Turk: Getting Started Guide

Table of Contents

Amazon Mechanical Turk Getting Started Guide

> ⚠️ **Important**
>
> This guide is deprecated. To get started using Mechanical Turk, we recommend you use the [Amazon Mechanical Turk Developer Guide](#).

This is the *Amazon Mechanical Turk Getting Started Guide*. This guide provides a conceptual overview of Amazon Mechanical Turk for developers who want to write scripts or software applications using the Amazon Mechanical Turk Requester API. The major sections of this guide are described in the following table. For information on using a graphical user interface to access Amazon Mechanical Turk functionality, go to the [Amazon Mechanical Turk Requester User Interface Guide](#).

Amazon Mechanical Turk is a web service that provides an on-demand, scalable, human workforce to complete jobs that humans can do better than computers, such as recognizing objects in photographs. For more information about this product, go to the Amazon Mechanical Turk [website](#).

How Do I...?

How do I...?	Relevant Topics
Get a general product overview of Mechanical Turk	[Introduction to Amazon Mechanical Turk](#)
Set up the accounts and tools needed to use Mechanical Turk	[Setting Up Accounts and Tools](#)
Use the Mechanical Turk SDKs to write and publish a Human Intelligence Task (HIT)	[Creating a HIT](#)
Learn more about implementing Mechanical Turk	[Implementing Amazon Mechanical Turk](#)

How do I...?	Relevant Topics
Find answers to common questions about Mechanical Turk	[Amazon Mechanical Turk FAQ](#)
Learn more about programmatically interacting with Mechanical Turk	[Amazon Mechanical Turk Developer Guide](#)
Find detailed information about Mechanical Turk operations	[Amazon Mechanical Turk API Reference](#)

Introduction to Amazon Mechanical Turk

The following topics provide a high-level overview of the Amazon Mechanical Turk web service. After reading these topics, you should understand the basics you need to work through the examples in this guide.

Topics

- [Overview of Amazon Mechanical Turk](#)
- [Key Amazon Mechanical Turk Concepts](#)
- [What's Next?](#)

Overview of Amazon Mechanical Turk

Amazon Mechanical Turk provides an on-demand, scalable, human workforce to complete jobs that humans can do better than computers. Amazon Mechanical Turk software formalizes job offers to the thousands of Workers willing to do piecemeal work at their convenience. The software also retrieves work performed and compiles it for you, the Requester, who pays the Workers for satisfactory work (only). Optional qualification tests enable you to select competent Workers.

The kinds of tasks humans can complete better than computers includes finding objects in photos, writing reviews of restaurants, movies, or businesses, translating text passages into foreign languages, getting the hours of operation of the business center within a hotel, determining if a hotel is family-friendly, or telling you the most relevant search results for a given phrase.

This guide presents a very slim slice of the Amazon Mechanical Turk API. For a complete description of the entire API, go to the [Amazon Mechanical Turk API Reference](#). For more information about using the API, go to the [Amazon Mechanical Turk Developer Guide](#).

Features

The following list describes the features of Amazon Mechanical Turk highlighted by the tutorial in this guide.

- **On-demand workforce—**Amazon Mechanical Turk provides access to a virtual community of Workers.
- **Create jobs that Workers perform over the Internet—**Advertise your job to the thousands of Amazon Mechanical TurkWorkers around the world

You prescribe the job (HIT) that Workers complete using their computer, and pay them for their work.

- **Test and publish your jobs—**Test your jobs in the Amazon Mechanical Turk sandbox and publish the revised jobs to the outside world.

Amazon Mechanical Turk provides SDKs and command line tools to make it easier to build solutions leveraging Amazon Mechanical Turk.

Key Amazon Mechanical Turk Concepts

This section describes the concepts and terminology you need to understand to use Amazon Mechanical Turk effectively.

Requester

A *Requester* is a company, organization, or person that creates and submits tasks (HITs) to Amazon Mechanical Turk for Workers to perform. As a Requester, you can use a software application to interact with Amazon Mechanical Turk to submit tasks, retrieve results, and perform other automated tasks. You can use the [Requester website](#) to check the status of your HITs, and manage your account.

Human Intelligence Task

A *Human Intelligence Task (HIT)* is a task that a Requester submits to Amazon Mechanical Turk for Workers to perform. A HIT represents a single, self-contained task, for example, "Identify the car color in the photo." Workers can find HITs listed on the Amazon Mechanical Turk website. For more information, go to the [Amazon Mechanical Turk website](#).

Each HIT has a lifetime, specified by the Requester, that determines how long the HIT is available to Workers. A HIT also has an assignment duration, which is the amount of time a Worker has to complete a HIT after accepting it.

Worker

A *Worker* is a person who performs the tasks specified by a Requester in a HIT. Workers use the [Amazon Mechanical Turk website](#) to find and accept assignments, enter values into the question

form, and submit the results. The Requester specifies how many Workers can work on a task. Amazon Mechanical Turk guarantees that a Worker can work on each task only one time.

Assignment

An *assignment* specifies how many people can submit completed work for your HIT. When a Worker accepts a HIT, Amazon Mechanical Turk creates an assignment to track the work to completion. The assignment belongs exclusively to the Worker and guarantees that the Worker can submit results and be eligible for a reward until the time the HIT or assignment expires.

Reward

A *reward* is the money you, as a Requester, pay Workers for satisfactory work they do on your HITs.

What's Next?

The next section explains how to sign up for AWS and for Amazon Mechanical Turk, and how to set up your development environment.

Setting Up Accounts and Tools

The following topics describe the tasks you need to perform before you can use Amazon Mechanical Turk.

Topics

- [Step 1: Sign Up for an AWS Account](#)
- [Step 2: Create a Requester Account](#)
- [Step 3: Link Your AWS account to your MTurk Requester account](#)
- [Step 4: Create an IAM User](#)
- [Step 5: Set up the Developer Sandbox](#)
- [Step 6: Set up an AWS SDK](#)
- [Step 7: Prepay for Your HITs or Enable AWS Billing](#)
- [Creating a HIT Tutorial](#)

Step 1: Sign Up for an AWS Account

To develop solutions using the Amazon Mechanical Turk web service, you must first sign up for an AWS account. An AWS account is an Amazon.com account that enables you to use services from AWS.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.

2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform [tasks that require root user access](#).

Using your AWS account, you can view your AWS account activity and usage reports and manage your security credentials.

AWS Security Credentials

AWS uses access keys to help protect your data when you access AWS programmatically. An access key consists of two parts: an access key ID, which is similar to a user name, and a secret access key, which is similar to a password.

- Sample access key ID: AKIAIOSFODNN7EXAMPLE

- Sample secret access key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

You use access keys to sign programmatic requests that you make to the Amazon Mechanical Turk API using a supported AWS SDK. You can use these access keys in both the sandbox and the production environment.

> ⓘ **Note**
>
> Although you may use your AWS *root credentials* to access the Mechanical Turk API, we recommend that instead you create an IAM *user* and use those credentials instead. Please note that Mechanical Turk does not currently support the use of IAM role credentials.

To manage your AWS security credentials

1. Sign in to the Amazon Web Services website at http://aws.amazon.com/security-credentials.

2. Do one of the following:

   - If you signed in using your root credentials, choose **Continue to Security Credentials**.

   - If you signed in as an IAM user, choose **Access Keys (Access Key ID and Secret Access Key)**.

3. Choose **Create New Access Key**.

4. In **Create Access Key**, choose **Download Key File**.

5. The private key file (`rootkey.csv`) is automatically downloaded by your browser. Save the private key file in a safe place.

> ⚠️ **Important**
>
> This is the only chance for you to save the private key file. You will need these credentials to use the Amazon Mechanical Turk API.

Step 2: Create a Requester Account

Before you can use Amazon Mechanical Turk, you must have an Amazon Mechanical Turk Requester account.

To create and register a Requester account

1. Go to https://requester.mturk.com.
2. Click **Create an Account**.
3. Enter your email address.
4. Follow the prompts to complete your Requester account registration.

   When prompted, type your mailing address and accept the **Amazon Mechanical Turk Participation Agreement**.

Step 3: Link Your AWS account to your MTurk Requester account

Next, you will need to link your AWS Account to your MTurk Requester Account. This operation grants permission to your AWS Account to access your Requester account using the MTurk APIs.

1. Go to https://requester.mturk.com/developer.
2. Click **Link your AWS Account** and sign-in with your AWS Root user email address and password.

Step 4: Create an IAM User

With AWS Identity and Access Management (IAM) you can securely control access to AWS services and resources for your users. Creating an IAM user allows you to use these credentials to access

the Amazon Mechanical Turk API instead of using your [root account credentials](). Please note that Mechanical Turk does not currently support the use of IAM role credentials.

> **ⓘ Note**
>
> You use IAM credentials only to authenticate Mechanical Turk API requests. You cannot use them to log in to the Mechanical Turk Requester website.

To create an IAM user

1. Sign in to the Amazon Web Services website at http://aws.amazon.com/security-credentials.

2. Choose **Get Started With IAM Users**.

3. Choose **Add user**.

4. Under **Set user details**, type the name for your user. To add multiple users with the same access types and permissions, click the **Add another user** link to enter another user name to create.

5. Under **Select AWS access type**, make sure **Programmatic access** is selected and choose **Next: Permissions** to move to the next step.

6. Under **Set permissions**, choose **Attach existing policies directly**.

7. Select the **AmazonMechanicalTurkFullAccess** or **AmazonMechanicalTurkReadOnly** policy, depending on the level of control you want to grant.

8. Choose **Next: Tags** to continue.

9. There is no need to add tags. Choose **Next: Review** to continue.

10. Review the settings you selected and choose **Create user** or **Create users** as appropriate.

11. The user is added. To download the user's credentials for use with SDKs or the CLI, either choose **Download .csv** to download the information of the user or users created. Alternatively, click the **Show** link next to the obscured **secret access key**. Copy it somewhere secure as you will only be able to view it once.

To set up Amazon Mechanical Turk permissions for previously created IAM users

1. Read [Overview of IAM Policies]() in the *IAM User Guide.*

2. Sign in to the IAM console at https://console.aws.amazon.com/iam/home#policies.

3.   On the **Policies** page, choose **AmazonMechanicalTurkFullAccess** or
     AmazonMechanicalTurkReadOnly.

4.   At the top of the page, choose **Policy Actions** and then choose **Attach**.

5.   On the **Attach Policy** page, select the IAM user(s) that you want to set permissions for. Then
     choose **Attach Policy** to give those users permission to access the Mechanical Turk API.

What is supported?

API authentication using IAM credentials	Yes
Website authentication using IAM credentials	No
Action-level permissions for IAM users	Yes
Resource-level permissions for IAM users	No
Tag-based permissions for IAM users	No
Authentication using temporary credentials (roles or groups)	No
Authentication using federated user credentials	No

Step 5: Set up the Developer Sandbox

You should test your HITs in the Amazon Mechanical Turk sandbox testing environment to make
sure they work as expected before publishing your HITs in the Mechanical Turk marketplace. The
sandbox provides a testing environment where you can publish and work on HITs to try them out
before publishing them in the Amazon Mechanical Turk Marketplace. The sandbox consists of a
Requester sandbox website and a Worker sandbox website.

You will need to create a Requester account on the Requester sandbox website, which is located at
https://requestersandbox.mturk.com. You'll also need to create a Worker account on the Worker
sandbox website located at https://workersandbox.mturk.com to view your sandbox created HITs
as a Worker. There is no charge for using the Mechanical Turk sandbox sites.

> ⓘ **Note**
>
>   To create HITs in the sandbox using the MTurk APIs, you will also need to link your
>   AWS account to your sandbox Requester account as per Link Your AWS account
>   to your MTurk Requester account, on the Requester sandbox website at https://
>   requestersandbox.mturk.com/developer.

When testing in the sandbox, test each HIT in all the major Internet browsers to make sure your HIT
works the same in each browser. If you have links to pictures or videos in your HITs, make sure the
links work. Also, your testing should include verifying that the format of the submitted answers is
acceptable.

Step 6: Set up an AWS SDK

The MTurk API can be accessed using the following AWS SDKs: Python/Boto (Boto3), Javascript
(NodeJS or Browser), Java, .NET, Go, Ruby, PHP or C++.

Configure the AWS SDK to use the 'us-east-1' region. This is the region in which the MTurk
API is available. Then connect to the MTurk Developer Sandbox and check your account
balance (the Sandbox should always return a balance of $10,000). To connect to the MTurk
Developer Sandbox, set the API endpoint in your SDK to **https://mturk-requester-sandbox.us-
east-1.amazonaws.com**. You can find examples here.

Step 7: Prepay for Your HITs or Enable AWS Billing

Before you use the production version of Amazon Mechanical Turk you need to prepay for the HITs
you create. Otherwise, you can't post your HITs to Workers.

To post your HITs to Workers, you must have money in your Prepaid HIT Balance to prepay for all of
your HITs. You can use a credit or debit card or debit card to prepay for the HITs.

Alternatively, you can enable AWS Billing for your account. You will see rewards and fees associated
with your HITs on the AWS Anniversary Bill for your linked AWS account.

For instructions on how to prepay for your HITs or to enable AWS Billing, go to the Requester
website.

Creating a HIT Tutorial

The Creating a HIT tutorial in the next section takes you step-by-step through using Amazon Mechanical Turk to create a HIT.

Creating a HIT

In the previous section, you set up your AWS account and installed the programming tools you need to use Amazon Mechanical Turk. The following topics describe how to write, test, publish, and manage a Human Intelligence Task (HIT).

Topics

- [Workflow](#)

Workflow

The following procedure gives you an overview of creating, testing, publishing, and managing a HIT.

Workflow for Requesters

1	Define your HIT.
	Construct your question in one of the [Question and Answer Data Structure](#). The Question paramater accepts a XML string or HTML string.

2	Create HIT.
	Build a new HIT with the [CreateHIT](#) operation or [CreateHITwithHITType](#) operation.
	Provide [Title, Description, Keywords, and Question Details](#) as outlined in the documentation
	For code samples demonstrating how to use the *Amazon Mechanical Turk Requester API*, go to [AWSLabs on Github.](#)
	For language-specific support about using *Amazon Mechanical Turk Requester API* go to [AWS SDKs](#)

3	Test your HIT.

Publish your HIT on the *Amazon Mechanical Turk Developer Sandbox*. The *Amazon Mechanical Turk Developer Sandbox* is a simulated environment that allows you to view your HIT as it would appear to Workers.

For more information about the *Amazon Mechanical Turk Developer Sandbox* and how to use it, go to the Amazon Mechanical Turk Developer Sandbox.

4	Publish your HIT on the Amazon Mechanical Turk production system.
	This step makes your HIT available to Workers.

5	Workers accept your HIT and complete the assignment.
	You can view the status of your HITs with AWS Shell or the AWS CLI.

6	Process the assignment results.
	When a Worker completes an assignment, you can view the results, output the results to a file, and accept or reject the work. Accepting the work means that you agree to pay the Worker.
	Use List Assignments for a HIT to get results once Workers have completed the Assignment. Then you can process the results with the ApproveAssignment operation and RejectAssignment operation.

7	Manage your HIT.
	You can extend the completion time for your HIT, expire the HIT early, add additional assignments, modify the HIT properties, or block Workers whose work does not meet your standards.

Implementing Amazon Mechanical Turk

The following topics describe information and additional resources you can use to implement a project using Amazon Mechanical Turk.

Topics

- [Interfaces](#)
- [Considerations](#)
- [Common Use Scenarios](#)
- [Coding Resources](#)
- [Advanced Functionality](#)
- [Reference Resources](#)

In the [Creating a HIT](#) tutorial you learned how to complete basic Amazon Mechanical Turk tasks. If you didn't use the tutorial, you can learn how to complete basic and advanced Amazon Mechanical Turk tasks using the *Amazon Mechanical Turk Developer Guide* and by looking at code samples. For more information, go to [Amazon Mechanical Turk Developer Guide](#) and [Amazon Mechanical Turk Sample Code and Libraries](#), respectively.

Interfaces

Amazon Mechanical Turk offers the following interfaces:

- Command line
- API
- Requester user interface

The Amazon Mechanical Turk command line interface (CLI) makes it easy to use most of the Amazon Mechanical Turk functionality. When you wish to have a hands-on approach to using Amazon Mechanical Turk and have a relatively small number of assignments and results, the CLI is a good choice. For more information, go to [Amazon Mechanical Turk Command Line Tool Reference](#).

When the number of assignments you have or the number of results you have is large, the Amazon Mechanical Turk API is a good choice. The API exercises all of Amazon Mechanical Turk's

functionality and enables you to integrate Amazon Mechanical Turk functions programmatically. For more information, go to Amazon Mechanical Turk Developer Guide.

If you have a very large number of similar HITs, consider using the Requester user interface. It merges one question template with lots of question data to create many similar HITs. For more information, go to Amazon Mechanical Turk Requester User Interface.

Considerations

Creating a successful HIT involves more than programming. There is a certain art involved, for example, in pricing a HIT correctly, laying out the question correctly, breaking down the task into HITs, and minimizing the Worker's time spent with the HIT. For that reason, we created a best practices guide that gives detailed instructions about creating an effective HIT.

Some of the HITs you create require the Workers to have special skills. You might, for example, publish a HIT that asks for a translation. Whenever you have a HIT that requires specialized skills, we recommend that you qualify the Workers. Only those who pass the test receive the opportunity to work on your HITs. For more information, go to CreateHIT and read about the `QualificationRequirement` parameter.

Once a Worker becomes qualified, you can grant them qualification status so that they no longer need to complete a qualification test before working on your HITs. For more information, go to AssignQualification.

Considerations for Writing a HIT

This guide presented a HIT for you to use. When you have to create your own, there are several things you should consider:

- What is the problem you are trying to solve?

  What questions do you want the Workers to answer? What is the best way to present the task to Workers? You need this information when you write the description and question for your HIT. You should be familiar with the format to create your question or task. For more information, go to the Amazon Mechanical Turk Developer Guide.
- How much do you want to pay Workers?

  You need to specify a reward for your HIT. Setting too low a reward discourages Workers from working on your HITs. You need to determine what a fair price is for the work you're asking the

Workers to do. The best way is to look at similar tasks advertised on the Amazon Mechanical Turk website. For more information, go to the [Amazon Mechanical Turk website](#).

- How many responses do you want?

  This is the number of assignments for your HIT. Sometimes you want only one answer per HIT. When the answer is controversial, however, you might like to get multiple answers (and thus use multiple assignments) per HIT and reach an answer by consensus.

- How much time do you want to allow to complete the task?

  Giving a Worker too long a time potentially delays getting results. Making the duration too short frustrates Workers.

Common Use Scenarios

This section describes some of the ways you can use Amazon Mechanical Turk.

Photo and Video Processing

Amazon Mechanical Turk is well-suited for processing images. In the past, companies have used Mechanical Turk to:

- Tag objects found in an image for easier searching and advertising targeting
- Select from a set of images the best picture to represent a product
- Audit user-uploaded images for inappropriate content
- Classify objects found in satellite imagery

Data Verification and Clean-up

Companies with large online catalogs use Amazon Mechanical Turk to identify duplicate entries and verify item details. Examples include:

- De-duplication of yellow pages directory listings
- Identification of duplicate products in an online product catalog
- Verification of restaurant details, such as the phone number or hours of operation

Information Gathering

The Amazon Mechanical Turk workforce enables you to gather information, such as:

- Allowing people to ask questions from a computer or mobile device about any topic and have Workers return results to those questions

- Filling out survey data on a variety of topics

- Writing reviews, descriptions and blog entries for websites

- Finding specific fields or data elements in large legal and government documents

Data Processing

Use Amazon Mechanical Turk to process data, for example:

- Podcast editing and transcription

- Human powered translation services

- Search engine rating

Coding Resources

To help you code your applications, we provide the following resources:

- **Developer Resources page—** Click the **Developer** tab on the Requester website located at https://requester.mturk.com/ to get to the Developer Resources page, which has links to sample code, documentation, the sandbox, and other helpful information.

- **Sample Code and Libraries—** You can use code samples as a means of understanding how to implement the Amazon Mechanical Turk API. For more information, go to Amazon Mechanical Turk Sample Code and Libraries.

- **Customer Forum—**We recommend you look at the Amazon Mechanical Turk Forum to get an idea of what other users are doing and to benefit from the questions they've asked.

  The forum can help you understand what you can and can't do with Amazon Mechanical Turk. The forum also serves as a place for you to ask questions that other users or Amazon representatives might answer. You can use the forum to report issues with the service, the API, or the documentation. For more information, go to Amazon Mechanical Turk Discussion Forums.

Advanced Functionality

The tutorial in this guide showed how to accomplish the basic tasks of creating, testing, and publishing your HITs. The Amazon Mechanical Turk API, however, offers advanced functionality, which the following table summarizes. For more information, go to the *Amazon Mechanical Turk Developer Guide*.

Functionality	Description
Review the work submitted by Workers	You reward work well done by paying the Worker. If work is not done well, you can elect not to pay the Worker and block them from future assignments.
Give bonuses	You might like to give a bonus to a Worker that does especially good work. This gives the Worker incentive to work on one of your other HITs. By rewarding Workers, you can create a small group of qualified people who you know do good work. They, in turn, will prioritize your HITs over others so that Workers complete your HITs in a timely manner. For more information about awarding bonuses, go to SendBonus.
Review results programmatically	Reviewing the results manually is an option when the number of assignments is small. As the number of assignments grows, reviewing results programmatically becomes more practical. The Amazon Mechanical Turk API enables you to do that. For more information, go to ListReviewableHITs
Delete a HIT	There are times when the results, although accurate, are unexpected and unusable by you. In that case, you have to delete the HIT, revise, and recreate it. For more information, go to DeleteHIT.
Update HIT properties	You can use API operations and the command line tools to update HIT properties, such as Title, Description, Reward, and Keywords. For more information, go to UpdateHITTypeOfHIT.
Extend or eliminate HITs	You can also use the API operations to extend the completion time for a HIT, expire a HIT early, or add additional assignments. For

Functionality	Description
	more information, go to [CreateAdditionalAssignmentsForHIT](#) and [DeleteHIT](#).
Communicate with Workers	You can use [NotifyWorkers](#) to send emails to specified Workers.
Return the amount of money in your account that you use to pay Workers	If you have not enabled AWS Billing, you must have sufficient Prepaid HITs balance for all assignments before you can publish Amazon Mechanical Turk HITs. You can programmatically access your Prepaid HITs balance using [GetAccountBalance](#).
Create your own qualifica tions for Workers	We recommend that you use qualification tests when completin g your HITs requires specialized skills. You can create your own qualification tests using [CreateQualificationType](#).

Reference Resources

The following list shows additional resources you can use to further your understanding of Amazon Mechanical Turk.

- Learn the pricing for Amazon Mechanical Turk.

  For more information, go to [Amazon Mechanical Turk Pricing](#).

- When you have questions about developing with Amazon Mechanical Turk visit the customer forum to get your questions answered.

  A wide variety of questions have already been answered in the forum. If your question has not been answered, ask it and wait for a fellow developer or Amazon representative to help. For more information, go to [Amazon Mechanical Turk Developer Forums](#).

- The Service Health Dashboard shows you the status of the Amazon Mechanical Turk web service. The dashboard shows you whether Amazon Mechanical Turk's services are functioning properly. For more information, go to the [Amazon Mechanical Turk Service Health Dashboard](#).

- The *Amazon Mechanical Turk Developer Guide* provides a detailed discussion of the service.

  For more information, go to [Amazon Mechanical Turk Developer Guide](#).

- The HIT page shows the Amazon Mechanical Turk HITs available to work on.

  For more information, go to the [HIT page](#).