

CLOUD ACCOUNT SETUP

creation of AWS Account:

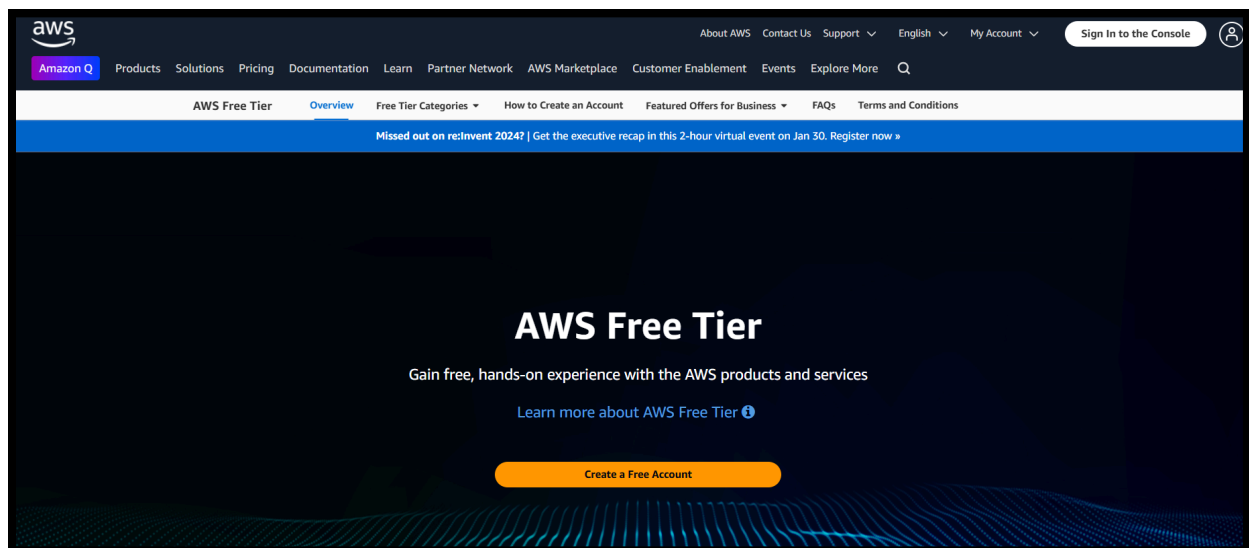
Here are some steps to create an AWS account,

step-1: Visit the AWS website:

- Open your web browser and go to the official AWS website:

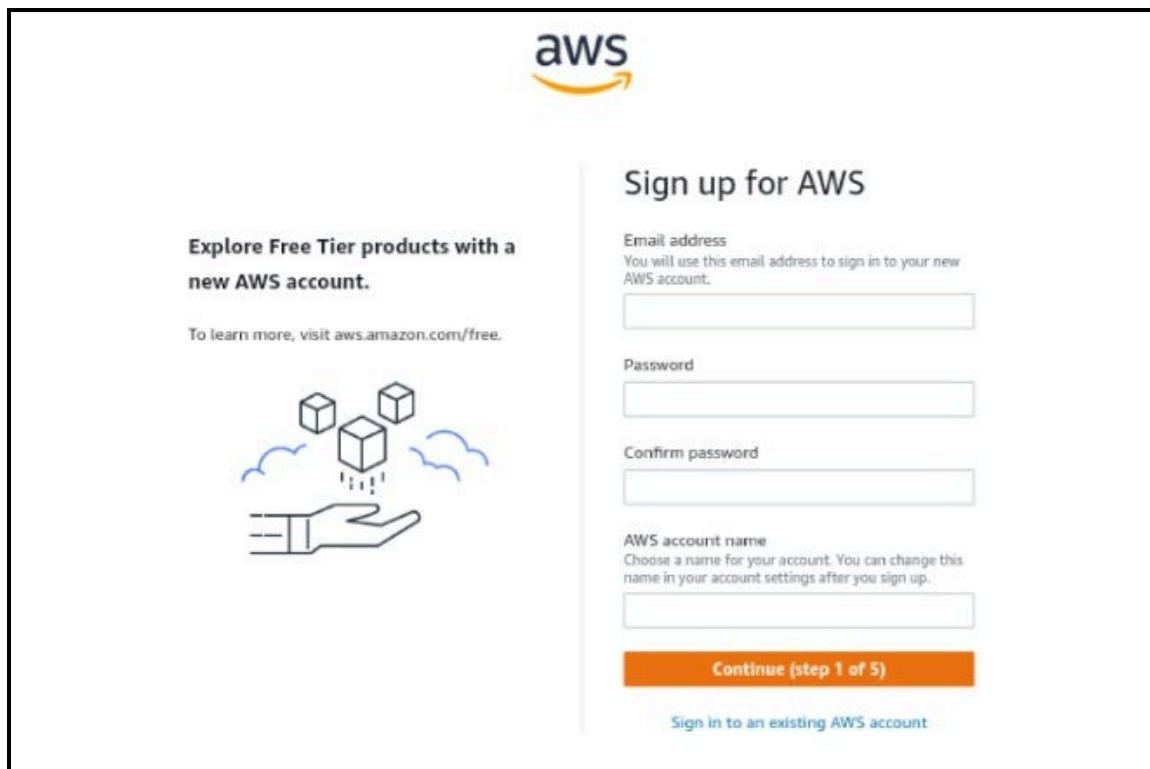
<https://aws.amazon.com/>

- Click on “create an AWS Account” button.



step-2: Enter Email and Account Name:

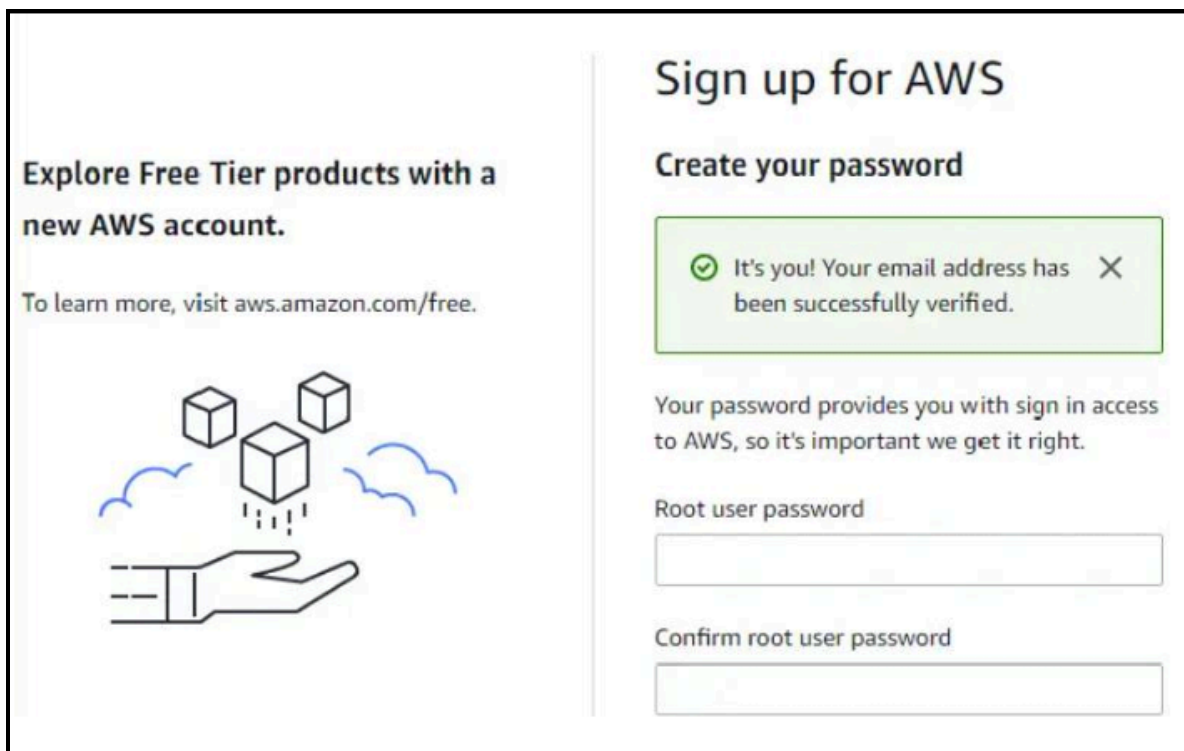
- Provide your Email Address.
- Choose an Account name.
- Click “verify Email Address”.



The image shows the AWS sign-up page. At the top center is the AWS logo. On the left side, there is a section titled "Explore Free Tier products with a new AWS account." with a link "To learn more, visit aws.amazon.com/free." Below this is an illustration of a hand holding three cubes. On the right side, the heading "Sign up for AWS" is followed by several input fields: "Email address" (with a subtext "You will use this email address to sign in to your new AWS account."), "Password", "Confirm password", and "AWS account name" (with a subtext "Choose a name for your account. You can change this name in your account settings after you sign up."). Below these fields is an orange "Continue (step 1 of 5)" button and a link "Sign in to an existing AWS account".

step-3:Verify Email:

- Check your Email inbox for a verification code from AWS.
- Enter the code on AWS Website and click verify.




The image shows the second step of the AWS sign-up process. The heading "Sign up for AWS" is followed by "Create your password". A green success message box states: "It's you! Your email address has been successfully verified." Below this, a subtext reads: "Your password provides you with sign in access to AWS, so it's important we get it right." There are two input fields: "Root user password" and "Confirm root user password".

step-4: Set root user password:

- Create a strong password for your root user account, This is the main account for your AWS.
- Click on continue.

Explore Free Tier products with a new AWS account.

To learn more, visit aws.amazon.com/free.



Sign up for AWS

Root user email address
Used for account recovery and some administrative functions

AWS account name
Choose a name for your account. You can change this name in your account settings after you sign up.

Verify email address

OR


[Sign in to an existing AWS account](#)

step-5:.Provide Account Information:


- Select “Business” or “Personal” account type.
- Enter your Business or Personal information.(name,mobile number,address)

Free Tier offers


All AWS accounts can explore 3 different types of free offers, depending on the product used.



Always free
Never expires



12 months free
Start from initial sign-up date



Trials
Start from service activation date

Sign up for AWS

Contact Information

How do you plan to use AWS?

☐ Business - for your work, school, or organization

☐ Personal - for your own projects

Who should we contact about this account?

Full Name

Phone Number

Country or Region

step-6:Accept the AWS customer Agreement:

- Review and accept the terms of AWS customer agreement.

Address

Apartment, suite, unit, building, floor, etc.

City


State, Province, or Region

Postal Code


☐ I have read and agree to the terms of the [AWS Customer Agreement](#).


step-7:Add a Payment Method:

- Enter your credit or debit card information.
- Click verify and continue.



Secure verification





 We will not charge you for usage below AWS Free Tier limits. We may temporarily hold up to \$1 USD (or an equivalent amount in local currency) as a pending transaction for 3-5 days to verify your identity.



Sign up for AWS

Billing Information

Credit or Debit card number



AWS accepts all major credit and debit cards. To learn more about payment options, review our [FAQ](#)

Expiration date

Cardholder's name

Billing address

☒ Use my contact address


☐ Use a new address

Verify and Continue (step 3 of 5)

You might be redirected to your bank's website to authorize the verification charge.

step-8:Verify your phone number:




- Choose how you want to receive a verification code(SMS or voice call).
- Enter the code you receive.



Sign up for AWS

Free Tier offers

All AWS accounts can explore 3 different types of free offers, depending on the product used.

-  **Always free**
Never expires
-  **12 months free**
Start from initial sign-up date
-  **Trials**
Start from service activation date

Contact Information

How do you plan to use AWS?

☐ Business - for your work, school, or organization

☐ Personal - for your own projects

Who should we contact about this account?

Full Name

Phone Number

Country or Region

Address
Apartment, suite, unit, building, floor, etc.

City

State, Province, or Region

Postal Code

☐ I have read and agree to the terms of the [AWS Customer Agreement](#).

Continue (step 2 of 5)

step-9:..Support Plan:

- Basic support plan is “free”.

step-10: complete sign up:

- Click on complete “sign up”.



Sign up for AWS

Select a support plan

Choose a support plan for your business or personal account. [Compare plans and pricing examples](#). You can change your plan anytime in the AWS Management Console.

☒ Basic support - Free

- Recommended for new users just getting started with AWS
- 24x7 self-service access to AWS resources
- For account and billing issues only
- Access to Personal Health Dashboard & Trusted Advisor



☐ Developer support - From \$29/month

- Recommended for developers experimenting with AWS
- Email access to AWS Support during business hours
- 12 (business)-hour response times



☐ Business support - From \$100/month

- Recommended for running production workloads on AWS
- 24x7 tech support via email, phone, and chat
- 1-hour response times
- Full set of Trusted Advisor best-practice recommendations



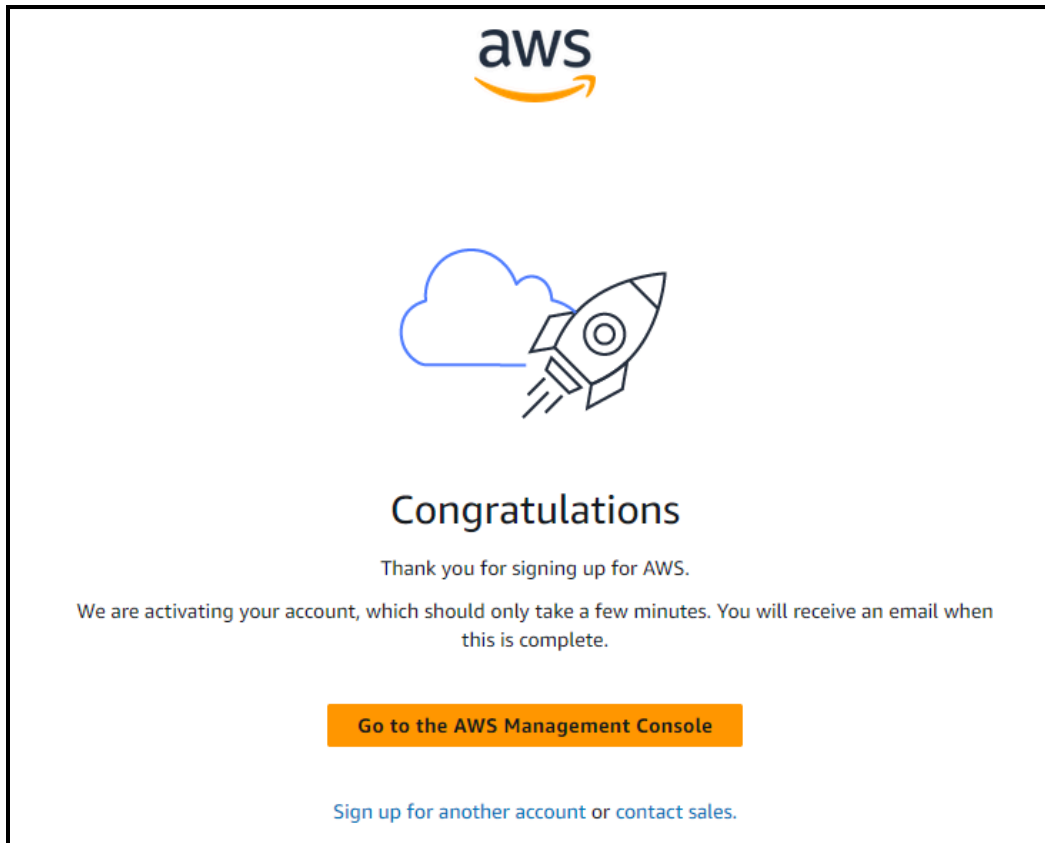
Need Enterprise level support?

From \$15,000 a month you will receive 15-minute response times and concierge-style experience with an assigned Technical Account Manager. [Learn more](#)

[Complete sign up](#)

step-11:Account Activation:

- Account will be activated in few minutes. Receives an Activation mail once the Activation is complete.

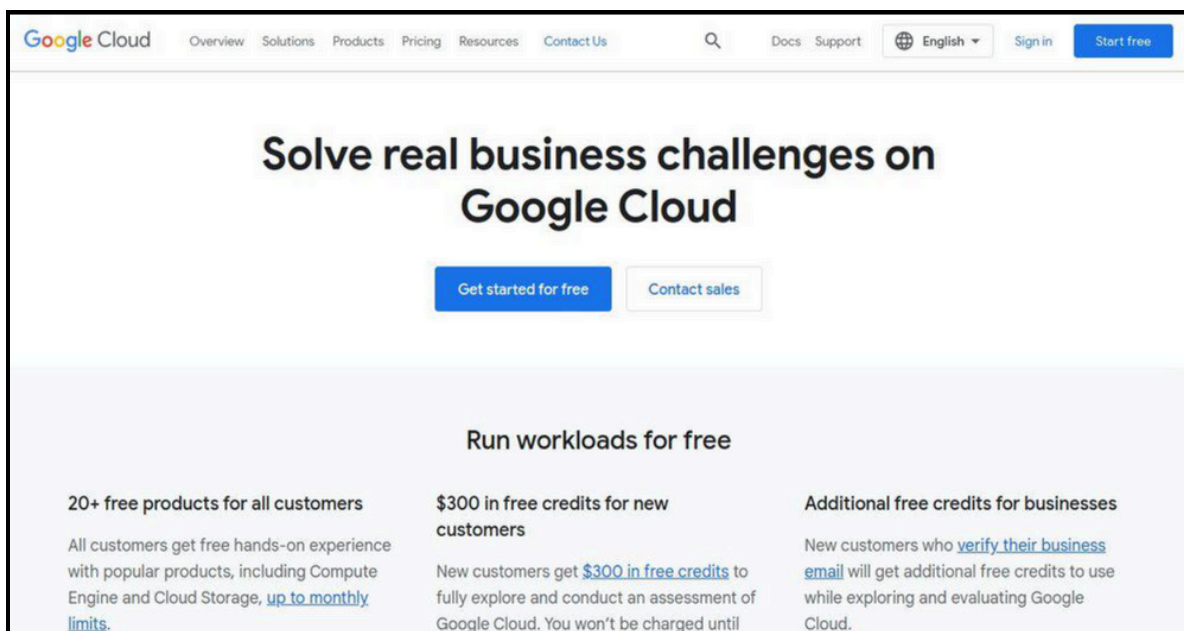


creation of GCP Account:

Here are some steps to create an GCP account,

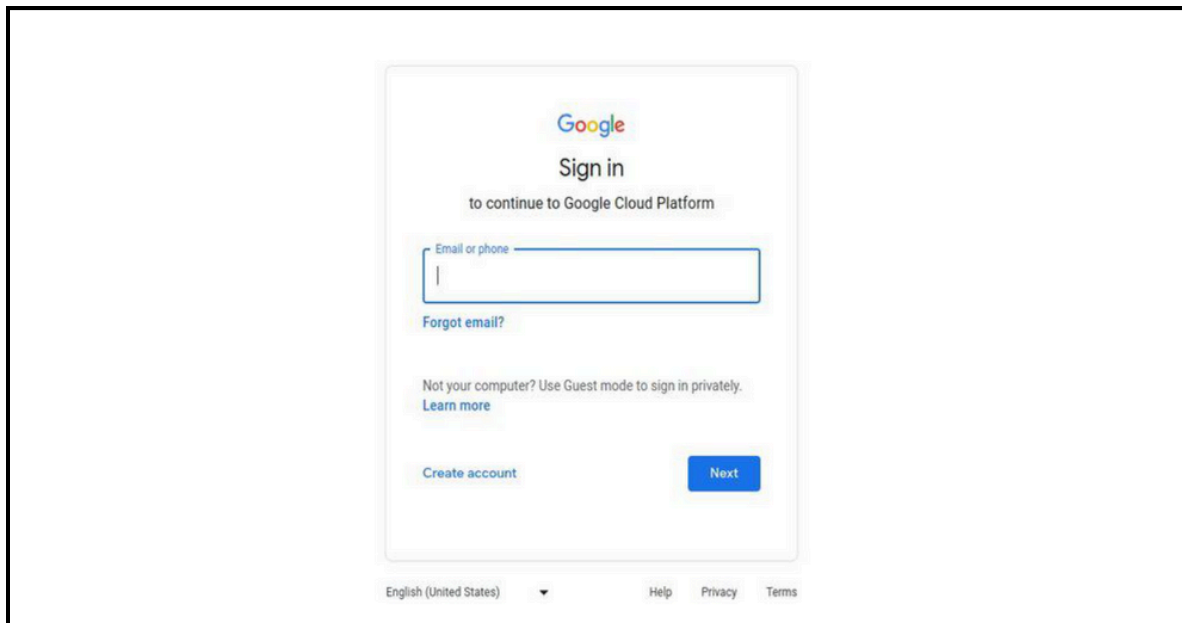
step-1:Go to the GCP account creation page:

- Open your web browser and navigate to the Google Cloud Console:
<https://console.cloud.google.com/>



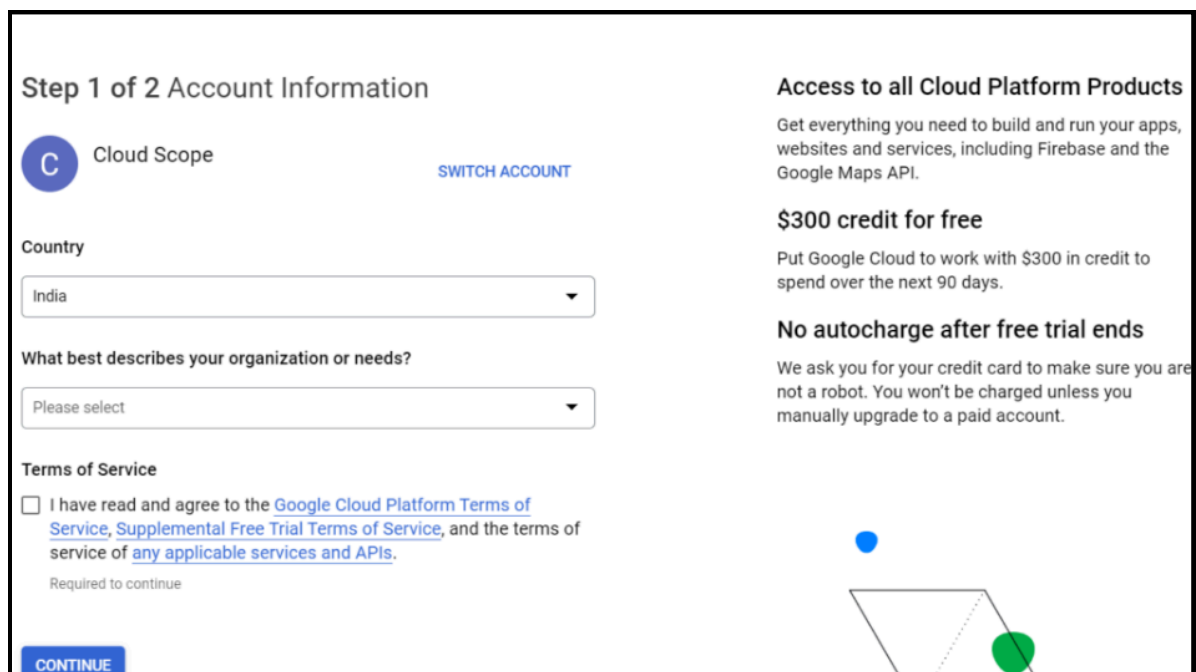
step-2:click on the start free button on the top corner.

step-3:sign in to your Gmail account:



step-4:Accept the GCP customer Agreement:

- Review and accept the terms of GCP agreement.


A screenshot of the "Step 1 of 2 Account Information" form for Google Cloud Platform. The form is titled "Step 1 of 2 Account Information" and includes a "Cloud Scope" logo and a "SWITCH ACCOUNT" link. The "Country" dropdown menu is set to "India". The "What best describes your organization or needs?" dropdown menu is set to "Please select". The "Terms of Service" section has a checkbox labeled "I have read and agree to the Google Cloud Platform Terms of Service, Supplemental Free Trial Terms of Service, and the terms of service of any applicable services and APIs." which is currently unchecked. Below the checkbox is the text "Required to continue". A blue "CONTINUE" button is at the bottom left. On the right side of the form, there is promotional text: "Access to all Cloud Platform Products" (Get everything you need to build and run your apps, websites and services, including Firebase and the Google Maps API.), "\$300 credit for free" (Put Google Cloud to work with \$300 in credit to spend over the next 90 days.), and "No autocharge after free trial ends" (We ask you for your credit card to make sure you are not a robot. You won't be charged unless you manually upgrade to a paid account.). A blue dot and a green circle are visible in the bottom right corner of the form area.

step-5:Enable Billing:

- Enter your debit card (only VISA and MASTER cards are applicable) or credit card information.

Step 2 of 2 Payment Information Verification


Your payment information helps us reduce fraud and abuse. You won't be charged unless you turn on automatic billing.

Account type 

Individual


Only Business accounts can have multiple users. You cannot change the account type after signing up. In some countries, this selection affects your tax options. [Learn more](#)

Payment method

Card number 

Card number is required

When billing starts, you'll be charged automatically each month.

 Reserve Bank of India requires that cards support automatic payments according to RBI regulations. If your card doesn't support automatic payments, you'll need to make manual payments or use a different card. We'll check your card in the next step. [Learn more](#)

Access to all Cloud Platform Products

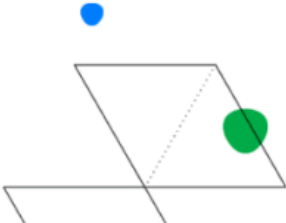
Get everything you need to build and run your apps, websites and services, including Firebase and the Google Maps API.

\$300 credit for free

Put Google Cloud to work with \$300 in credit to spend over the next 90 days.

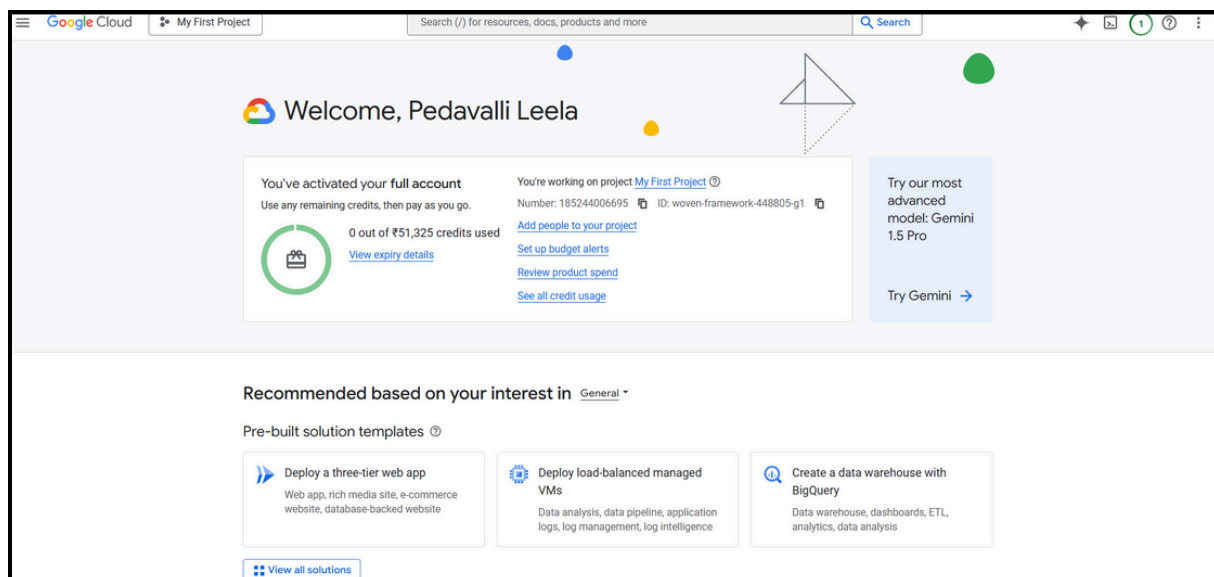
No autocharge after free trial ends

We ask you for your credit card to make sure you are not a robot. You won't be charged unless you manually upgrade to a paid account.



step-6:Account completion:

- After completion of payment you will be redirected to Gcp console page.



☐ Introduction To DevOps:

Write a brief summary (250-300 words) explaining the following:

- DevOps culture and principles:** What does it mean to have a DevOps culture in an organization?

DevOps is a set of practices that combines software development (Dev) and IT operations (Ops). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

A DevOps culture emphasizes collaboration, communication, and shared responsibility between development and operations teams.

Here's a breakdown of what it means to have a DevOps culture:

1. Collaboration and Communication:

- Development and operations teams work closely together, sharing knowledge, and communicating openly throughout the entire process.
- GCP fosters this collaboration through several integrated services, such as Cloud Source Repositories, Cloud Build, and Google Kubernetes Engine (GKE).

2. Automation:

- Automating tasks like testing, deployment, and infrastructure provisioning reduces manual effort, minimizes human error, and increases efficiency.
- This allows for faster and more frequent releases of software.

3. Continuous Improvement:

- Regularly analyzing and improving the development and delivery process based on feedback and data.
- This iterative approach ensures that the organization is constantly learning and adapting to improve its software delivery capabilities.

4. Continuous Integration/Continuous Delivery (CI/CD):

- Regularly integrating code changes into a shared repository and automating the build, test, and deployment process.

5. Shared Responsibility:

- Development and operations teams share responsibility for the entire software lifecycle, from development to production support.

6. Feedback:

- Continuous feedback is a core principle with rapid feedback from monitoring, testing, and deployments, teams can quickly address issues, which leads to improved software quality and user satisfaction.

☐ The DevOps lifecycle:

Outline the key phases of the DevOps lifecycle and their importance.

The DevOps lifecycle is a continuous process that involves several stages aimed operations to improve collaboration and efficiency. Here's a brief overview of the key stages in the DevOps lifecycle:

1. planning:

- Define project requirements and objectives.
- Create a road map for the project.

2. Development:

- Writing and Reviewing the code.
- Testing the code.

3. Testing:

- performs Automated testing to ensure code quality.
- Identify errors in the development process.

4. Continuous Integration:

- Runs Automated testing to ensure code quality.

5. Continuous Deployment:

- Continuous deployment to push changes to product quickly.

6. Monitoring:

- Monitor application performance.
- System health and optimization.

7. Feedback:

- Gathering feedback from users.
- use feedback for improvement in project.

☐ Benefits of DevOps in software development:

Discuss how adopting DevOps can improve software delivery and collaboration.

1. Efficiency:

- Automating repetitive tasks such as testing, deployment and minimize the risk of human resources.
- Efficiency leads to Cost savings and time saving.

2. Faster Updates:

- Continuous delivery practices enable quicker updates to applications.

3. **Higher quality software:**

- continuous testing improves software quality.

4. **Reliability:**

- The stable operation of software applications and services.

5. **Improved Collaboration:**

- Improves a collaboration between development and operation teams.
- Encouraging shared responsibility.

6. **Cost savings:**

- Efficient processes, fast recovery contribute to cost savings.

☐ Explore DevOps Tools:

Choose one tool from each category(CI/CD, Containerization, infrastructure as a code)and provide:

A brief overview (150-200 words) of the tool, its primary purpose, and how it fits into the DevOps lifecycle.

Examples of how this tool can be used on AWS, Azure, and GCP. Include links to the official documentation or relevant resources.

1.CI/CD: Jenkins

Description: Jenkins is an automation server that helps coders continuously integrate and deliver their code. It allows fast software releases with better quality of software due to the automation of the build, test, and deploy processes.

DevOps Lifecycle:

Jenkins is a critical component in the CI/CD pipeline. It automates the following stages:

Continuous Integration (CI):

Builds and tests code changes automatically as soon as they are committed to the version control system.

This makes it easier to catch integration problems early.

Continuous Delivery (CD):

Deploy validated code changes automatically to various environments (staging, production, etc.)

Examples

AWS:

Integrate with AWS Code Pipeline to build and manage CI/CD pipelines

Use AWS EC2 instances to run your Jenkins servers

Use AWS services like S3 to host your artifacts and Code Deploy to do deployments

Azure:

Enable use through Azure DevOps Services with CI/CD pipeline

Run Jenkins on Azure Virtual Machines or AKS.

c) Use of Azure Blob Storage to store artifacts, and deployment using Azure Pipelines.

GCP:

a) Google Cloud Build can be used for implementing the CI/CD pipeline.

b) Jenkins can be run on Google Compute Engine or GKE.

c) Use of Google Cloud Storage to store artifacts and Google Cloud Run to deploy the application.

2. Containerization: Docker:

Docker is an open-source platform for developing, shipping, and running a containerized application. It packages applications and their dependencies together inside containers so that the applications run uniformly regardless of their environment.

DevOps Lifecycle:

Docker aids in the following: Development-snowballing and testing software in isolated environments;

Deployment-deploying its applications easily on any environment-on-premises, cloud, or hybrid;

Scalability-scale down or up with demand.

Examples

AWS

Run Docker containers on Amazon ECS (Elastic Container Service) or Amazon EKS (Elastic Kubernetes Service).

Use AWS Fargate for container orchestration in a serverless approach.

Azure:

Run Docker containers on Azure Container Instances or Azure Kubernetes Service (AKS)

GCP:

Run Docker containers on Google Kubernetes Engine (GKE) or Google Cloud

Run.

3. Infrastructure as Code (IaC): Terraform

Terraform is a completely open source, infrastructure-as-code tool enabling one to describe and create both physical and virtualised cloud and network infrastructures through declarative configuration files.

DevOps Lifecycle: Terraform enables infrastructure provisioning. With the help of terraform, we can automate creation and management of the infrastructure resources.

Version control: Track infrastructure changes as one would code, allowing better collaboration and easier rollbacks.

Consistency: Ensure that infrastructure deployments are consistent across various environments.

Examples:

AWS:

Provision different services of AWS, including EC2, S3, RDS, etc.

Azure:

Set up a multi-tier application with VMs in different availability zones.

GCP:

Deploy a Kubernetes cluster on Google Kubernetes Engine (GKE).

☐ Tools Overview:

CI/CD Tools: Jenkins, GitHub Actions .

Containerization Tools: Docker, Kubernetes.

Infrastructure as Code Tools: Terraform, AWS CloudFormation, Azure Resource Manager (ARM) templates.

- **CI/CD tools**

- **Jenkins:** Jenkins is an automation server that provides totally free and opensource continuous integration and continuous delivery pipelines, auto-automates build, testing, and the deployment of the codes.
- **GitHub Actions:** It is a CI/CD platform natively built in GitHub, allowing

users to automate any software workflow, including builds, tests, and deployments, right inside their repositories.

- **Containerization tools**

- **Docker:** Docker is the platform through which users can create, execute, and manage applications using containers. Its Docker utility simplifies the process of packaging and deployment of an application because it isolates an application along with all of its dependencies into a container.

- **Kubernetes** is an open-source system that manages containerized applications across multiple clusters of hosts and automates their deployment, scaling, and management.

- **Infrastructure as Code Tools**

- **Terraform:** An open-source infrastructure as a code tool based on declarative language used in defining and providing infrastructure resources to include servers, network, and also storage.

- **AWS CloudFormation :** The service provisioned by the AWS in generating and provision AWS resources based on the description within JSON or YAML.

- **Azure Resource Manager (ARM) templates:** A service offered by Azure that enables users to model and provision Azure resources using templates written in JSON.