

Homework — The Cloud, Demystified

Estimated time: ~45 min · No code required · No paid resources created

Three short tool-tour exercises. Goal: turn the abstract picture from the theory page (IaaS / PaaS / SaaS, regions, the billing model) into something you've actually *clicked through* in a real cloud console. You will sign up for AWS Free Tier, compare the three major consoles side-by-side, and read a simulated cloud bill. Take a screenshot at each step.

Money rule — read this first

These tasks involve real cloud-provider sign-up flows. **Do not click "Launch", "Create", or "Deploy" on any actual paid resource.** The Free Tier is \$0 only if you don't exceed its limits — the easiest way to stay at \$0 is to never create the resource in the first place. You will need a real credit card for AWS sign-up; that's normal and expected. Close every tab before walking away. If you accidentally launch something, immediately go back and click "Terminate" / "Delete".

Task 1 — AWS Free Tier sign-up & the EC2 launch wizard (don't launch)

First we'll create an AWS account and walk to the edge of launching a virtual machine — *without* actually launching it. The goal is to see what the wizard asks you for, not to run anything.

1. Open aws.amazon.com/free and sign up. You'll need an email address, a credit card, and a phone number for verification. The Free Tier itself is \$0 if you stay inside its limits.
2. Once you land in the AWS Management Console, use the search bar at the top to find **EC2** and open it.
3. Click **Instances** in the left sidebar, then click the orange **Launch instances** button. The launch wizard appears.

4. **STOP** — do not click "Launch instance" at the bottom. Instead, scroll through the wizard and look at every section it asks you about: *Name, Application and OS Images (AMI), Instance type, Key pair (login), Network settings / Security group, Configure storage*. Notice the "*Free tier eligible*" green label next to certain AMI and instance-type choices.
5. Take a screenshot of the launch wizard showing those dropdowns.
6. Close the tab **without** clicking Launch. (If you did click Launch by accident: go back to *Instances*, select the instance, *Instance state* → *Terminate*.)

✓ **Self-check:** you can name **three things** the wizard asked you to pick (e.g., AMI, instance type, key pair, security group, storage size); you can point to where the "*Free tier eligible*" filter / label is in the AMI & instance-type sections; you understand that **nothing is running and nothing is being billed** because you never clicked the final Launch button.

Task 2 — Compare the three major consoles

Same primitive (a virtual machine), three different consoles. We'll open AWS, GCP, and Azure side-by-side and notice how each provider frames the same idea.

1. **AWS:** sign in to the console from Task 1. In the search bar, open **EC2** → **Instances**. This is AWS's "list of virtual machines" page.
2. **GCP:** open console.cloud.google.com. Any free Google account works — you don't need to enable billing or accept the \$300 free trial. Use the left/top menu to find **Compute Engine** → **VM instances**. (You may have to "create a project" first — that's free and doesn't create any chargeable resources.)
3. **Azure:** open portal.azure.com. A Microsoft account works. Use the top search bar to find **Virtual machines**. This is Azure's equivalent page.
4. Take a screenshot of each provider's compute landing page.
5. Write **one sentence per provider** noting one thing that looks *different* about how that console is organised. Examples: "*Azure groups everything into 'Resource Groups' in the left nav*", "*GCP forces you to pick a project from a header dropdown before anything shows up*", "*AWS's region selector is in the top-right and the whole console rebuilds when you change it*".

✓ Self-check: you can map the three names — **EC2 (AWS) = Compute Engine (GCP) = Virtual Machines (Azure)** — from memory; you have a one-sentence note for each provider about one organisational quirk; you understand that the underlying primitive ("a Linux VM you SSH into") is the same on all three — the marketing names and the menu structure are what differ.

Task 3 — Read a real cloud bill (using the calculator, \$0 real cost)

Cloud pricing is the part that surprises people. We'll model the same tiny workload in AWS's official price calculator and see what makes the bill move.

1. Open calculator.aws. Click **Create estimate**, search for **EC2**, and click *Configure*.
2. Configure these inputs (don't worry about the rest — defaults are fine):
 - **Region:** `US East (N. Virginia) us-east-1`
 - **Operating system:** Linux
 - **Instance type:** `t3.small`, running **24/7** (730 hours / month)
 - **Storage (EBS):** 50 GB, general purpose (`gp3`)
 - **Data transfer out to the internet:** 100 GB / month

Note the **estimated monthly cost**. Write it down.

3. Now change the region from `us-east-1` to `EU (Frankfurt) eu-central-1`. Note the new monthly cost. Write it down.
4. Now change **Data transfer out** from 100 GB to **1 TB** (1024 GB). Note the new monthly cost. Write it down.
5. Take a screenshot of the final estimate breakdown (showing compute, storage, and data-transfer line items separately).

✓ Self-check: you can answer (a) approximately how much an **idle** `t3.small` costs per month (compute + 50 GB storage + 100 GB egress, all in `us-east-1`); (b) **yes/no**: does region change the price for the same instance type? (c) when egress went from 100 GB to 1 TB, which line item *dominated* the bill — compute, storage, or data transfer? You now understand why "*the cloud is*

expensive" is usually a statement about **egress and idle resources**, not about compute per se.

What "done" looks like

You finish this homework when you have, in your notes:

- A screenshot of the EC2 launch wizard, plus three things it asked you to pick — without having launched anything (Task 1).
- Three screenshots (one per provider) of the compute landing page in AWS, GCP, and Azure, plus a one-sentence "what's different" note for each (Task 2).
- Three monthly cost numbers from the calculator (us-east-1 / 100 GB, eu-central-1 / 100 GB, eu-central-1 / 1 TB) and a one-line answer to "which line item dominated when egress went from 100 GB to 1 TB?" (Task 3).

That's the whole cloud picture from the theory page, observed in real consoles you can log into right now: *the primitive (a VM) is the same everywhere; the consoles look different; the bill is dominated by whatever you forgot to turn off — usually egress or idle resources*. Next time someone says "*just put it in the cloud*", you'll know what questions to ask first.