

## Part II #1

The screenshot shows the Google Cloud Compute Engine interface. The left sidebar is collapsed. The main header bar includes the Google Cloud logo, a project selector for 'My First Project', a search bar with placeholder 'Search (/) for resources, docs, products, and more', and a 'Search' button. To the right are icons for notifications (3), help, and user profile.

The main content area is titled 'VM instances' and features a 'Create Instance' button. Below the title are tabs for 'Instances', 'Observability', and 'Instance schedules'. A table lists a single VM instance named 'nginx'. The table columns include Status (green circle), Name (nginx), Zone (us-central1-c), Recommendations, In use by, Internal IP (10.128.0.2 (nic0)), External IP (35.226.34.34 (nic0)), Connect (SSH dropdown), and a more options menu. A 'Filter' input field is present above the table.

Below the table is a section titled 'Related actions' containing six cards:

- Explore Backup and DR** [New] - Back up your VMs and set up disaster recovery.
- View billing report** - View and manage your Compute Engine billing.
- Monitor VMs** - View outlier VMs across metrics like CPU and network.
- Set up firewall rules** - Control traffic to and from a VM instance.
- Patch management** - Schedule patch updates and view patch compliance on VM instances.
- Load balance between VMs** - Set up Load Balancing for your applications as your traffic and users grow.

A 'Hide' link is located at the top right of the related actions section.

## Part II #5

The screenshot shows a dark-themed web page with a white content area. The title 'Welcome to nginx!' is centered at the top. Below it is a paragraph of text: 'If you see this page, the nginx web server is successfully installed and working. Further configuration is required.' Another paragraph follows: 'For online documentation and support please refer to [nginx.org](http://nginx.org). Commercial support is available at [nginx.com](http://nginx.com)'. At the bottom, there is a small note: 'Thank you for using nginx.' followed by the signature 'derrowjb'.

### Part III #5

The screenshot shows the Google Cloud Compute Engine interface. The left sidebar has sections for Overview, Virtual machines (selected), Instance templates, Sole-tenant nodes, Machine images, TPUs, Committed use discounts, Reservations, and Migrate to Virtual Machines. The main content area shows a list of VM instances with columns for Status, Name, Zone, Recommendations, In use by, Internal IP, External IP, and Connect (SSH). Instances listed include hpcslurm-controller, hpcslurm-debugnodedset-0, hpcslurm-debugnodedset-1, hpcslurm-debugnodedset-2, hpcslurm-debugnodedset-3, and hpcslurm-slurm-login-001. Below the table are related actions like Explore Backup and DR, View billing report, Monitor VMs, Explore VM logs, Set up firewall rules, Patch management, and Load balance between VMs.

Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	<a href="#">hpcslurm-controller</a>	us-central1-a			10.0.0.2 (nic0)	34.71.89.113 (nic0)	SSH
<input type="checkbox"/>	<a href="#">hpcslurm-debugnodedset-0</a>	us-central1-a			10.0.0.5 (nic0)		SSH
<input type="checkbox"/>	<a href="#">hpcslurm-debugnodedset-1</a>	us-central1-a			10.0.0.6 (nic0)		SSH
<input type="checkbox"/>	<a href="#">hpcslurm-debugnodedset-2</a>	us-central1-a			10.0.0.4 (nic0)		SSH
<input type="checkbox"/>	<a href="#">hpcslurm-debugnodedset-3</a>	us-central1-a			10.0.0.7 (nic0)		SSH
<input type="checkbox"/>	<a href="#">hpcslurm-slurm-login-001</a>	us-central1-a			10.0.0.3 (nic0)	34.72.120.156 (nic0)	SSH

**Related actions**

- Explore Backup and DR New  
Back up your VMs and set up disaster recovery
- View billing report  
View and manage your Compute Engine billing
- Monitor VMs  
View outlier VMs across metrics like CPU and network
- Explore VM logs  
View, search, analyze, and download VM instance logs
- Set up firewall rules  
Control traffic to and from a VM instance
- Patch management  
Schedule patch updates and view patch compliance on VM instances
- Load balance between VMs  
Set up Load Balancing for your VM instances

### Part III #7

```
ssh.cloud.google.com/v2/ssh/projects/fifth-inwell-457000-n7/zones/us-central1-a/instances/hpcslurm-slurm-login-001?authuser=0&hl=en_US&projectNumber=517820908044&useAdminProxy=true - Google Chrome
ssh.cloud.google.com/v2/ssh/projects/fifth-inwell-457000-n7/zones/us-central1-a/instances/hpcslurm-slurm-login-001?authuser=0&hl=en_US&projectNumber=517820908044&useAdminProxy=true

SSH-in-browser
 UPLOAD FILE DOWNLOAD FILE

srun: error: Node failure on hpcslurm-computenodeset-4
srun: error: Nodes hpcslurm-computenodeset-(4-7) are still not ready
srun: error: Something is wrong with the boot of the nodes.
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 8 hostname
srun: error: Unable to allocate resources: Requested topology configuration is not available
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 4 hostname
hpcslurm-debugnode0set-0
hpcslurm-debugnode0set-1
hpcslurm-debugnode0set-2
hpcslurm-debugnode0set-3
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ module load mpi
lmod has detected the following error: The following module(s) are unknown: "mpi"
Please check the spelling or version number. Also try "module spider ...".
It is also possible your cache file is out-of-date; it may help to try:
$ module --ignore_cache load "mpi"

Also make sure that all modulefiles written in TCL start with the string #Module

[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ module load openmpi
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ curl -O https://w3.cs.jmu.edu/lam2mo/files/cluster/mpi_hello.c
% Total % Received % Xferd Average Speed Time Time Current
          Dload Upload Total Spent Left Speed
100 635 100 635    0     0 2783      0:--:--:--:--:-- 2772
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ mpicc -o hello mpi_hello.c
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -p compute -n 8 ./hello
srun: error: Node failure on hpcslurm-computenodeset-4
srun: error: Nodes hpcslurm-computenodeset-(4-7) are still not ready
srun: error: Something is wrong with the boot of the nodes.
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 8 ./hello
srun: error: Unable to allocate resources: Requested topology configuration is not available
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 4 ./hello
Supervisor: Rank 0 on hpcslurm-debugnode0set-0 (4 total processes)
  Message from rank 1 on hpcslurm-debugnode0set-1
  Message from rank 2 on hpcslurm-debugnode0set-2
  Message from rank 3 on hpcslurm-debugnode0set-3
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 8 ./hello
srun: error: Unable to allocate resources: Requested topology configuration is not available
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$ srun -n 4 ./hello
Supervisor: Rank 0 on hpcslurm-debugnode0set-0 (4 total processes)
  Message from rank 1 on hpcslurm-debugnode0set-1
  Message from rank 2 on hpcslurm-debugnode0set-2
  Message from rank 3 on hpcslurm-debugnode0set-3
[joshb12d_gmail_com]@hpcslurm-slurm-login-001 ~$
```

## Part IV #3

The screenshot shows the Google Cloud Platform Cloud Run service details page for a service named "cloud fun". The top navigation bar includes "My First Project", "Search" (with a search term "cloud fun"), and various user and project management icons. Below the navigation, the service name "cloud fun" is displayed along with "Edit & deploy new revision", "Set up Continuous Deployment", and "Test" buttons.

The main content area is titled "Deploying revision" and shows a list of deployment steps:

- Building source (see logs) Completed
- Updating service Completed
- Creating revision Completed
- Routing traffic Completed

Below this, the service revision is listed as "sha512" with a green checkmark, and the URL is "https://sha512-517820908044.us-central1.run.app". The scaling is set to "Auto (Min: 0)".

The "Source" tab is selected, showing the function entry point "hello\_http" and a "Edit source" button. The code editor displays the following Python code:

```
1 import functions_framework
2 import hashlib
3
4 #functions_framework.http
5 def hello_http(request):
6     if request.args and 'message' in request.args:
7         msg = request.args['message']
8         return hashlib.sha512(bytes(msg, 'utf-8')).hexdigest()
9     else:
10        return "Error"
11
12
```

Below the code editor, there is a "Show payload" link and a "Download ZIP" button.

## Part IV #5

3615f80c9d293ed7402687f94b22d58e529b8cc7916f8fac7fd7fb5af4cf777d3d795a7a00a16bf7e7f3fb9561ee9baae480da9fe7a18769e71886b03f15