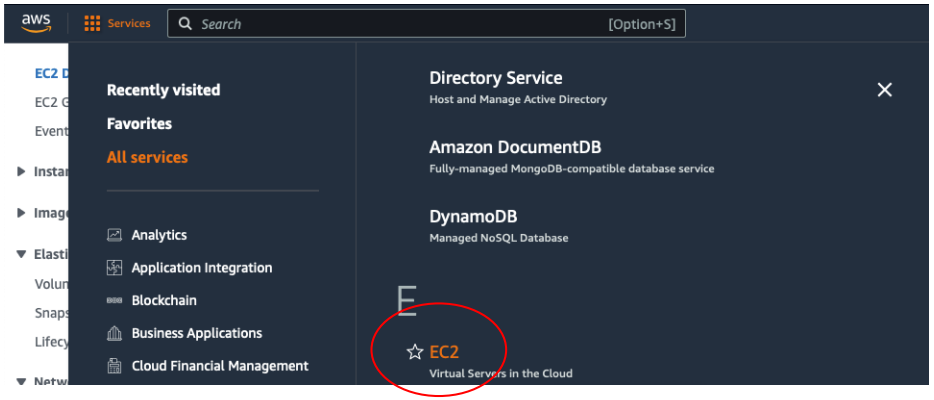
CPU Run on AWS

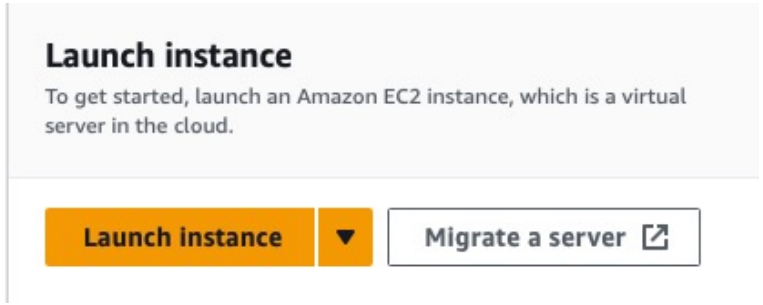
Programming FPGAs for Economics:
An Introduction to Electrical Engineering Economics

Bhagath Cheela, Alessandro Peri, André DeHon, Jesús Fernández-Villaverde

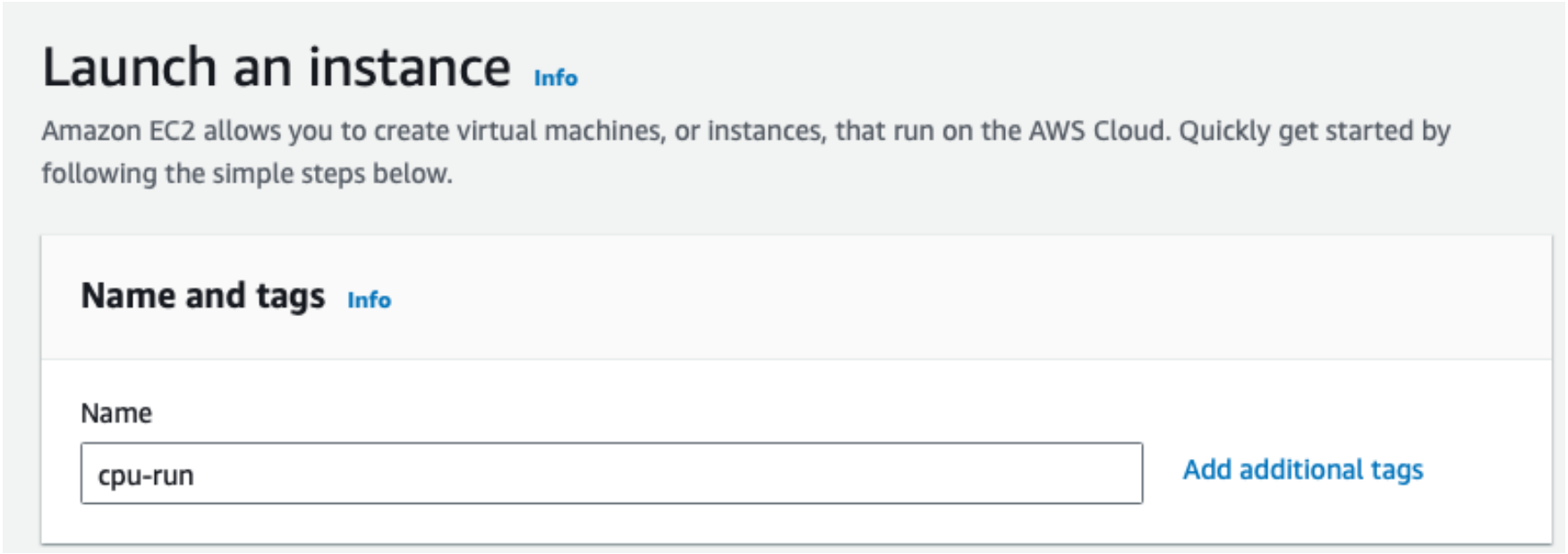
Steps



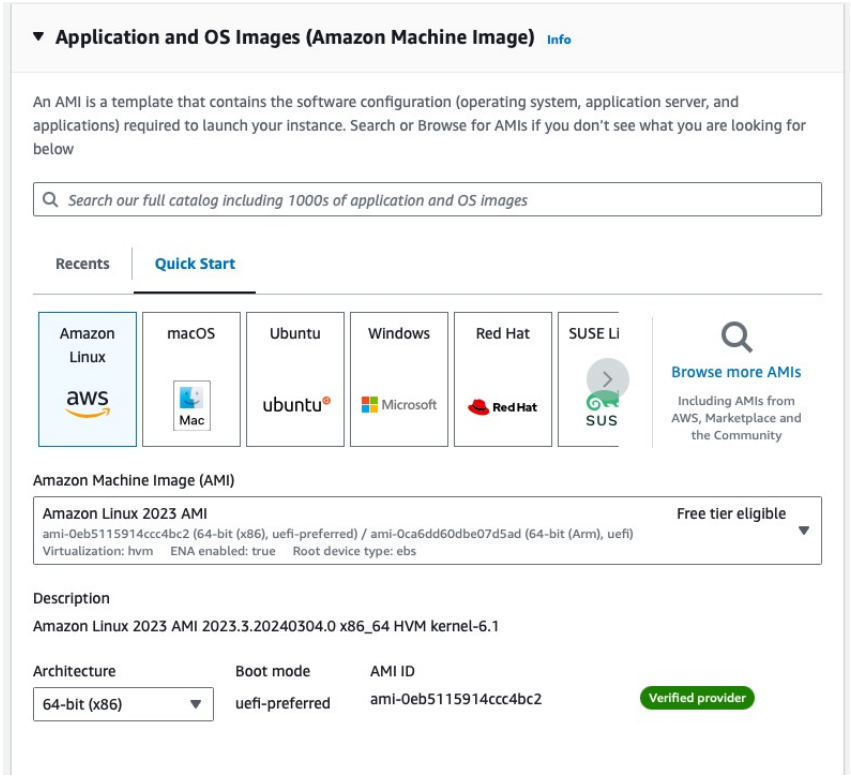
1. **Log into your AWS account:**
2. Navigate to the Home Console
3. Select **EC2**
4. Launch Instance



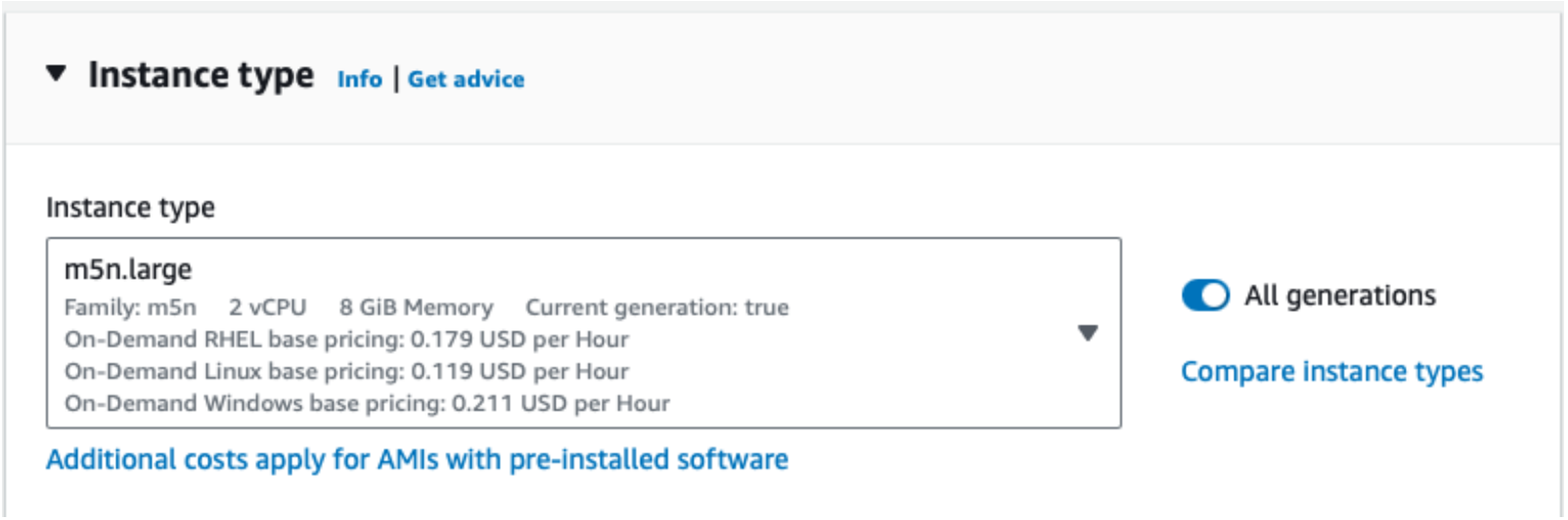
Steps: Name and tags



Select Amazon Linux Instance

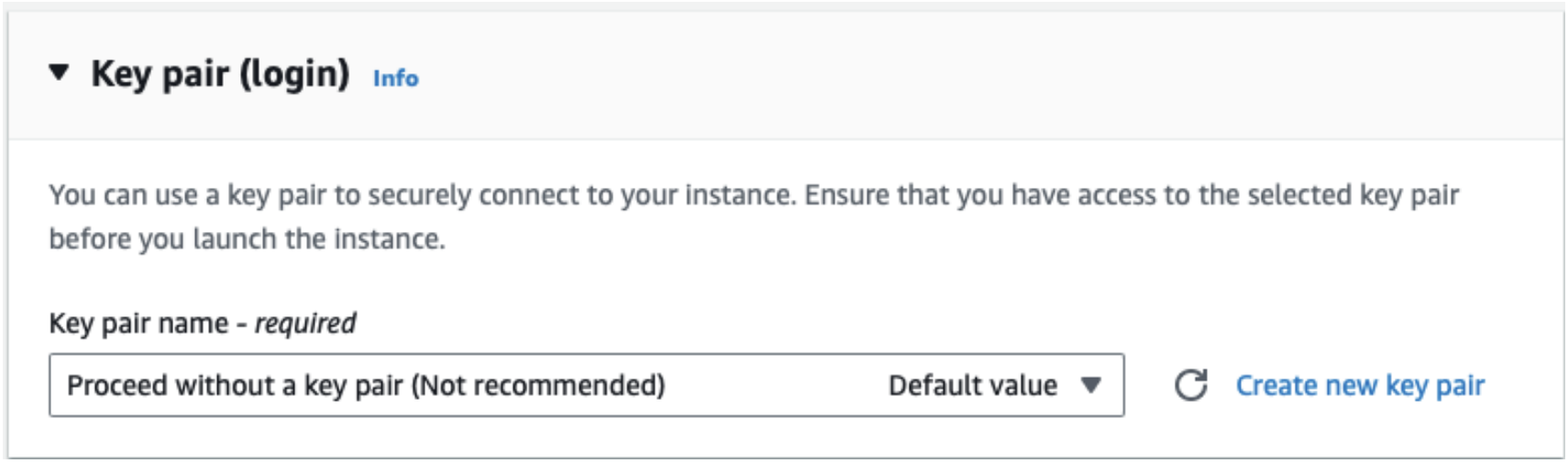


Select CPU Instance



Repeat the same for all other FPGA instances: m5n.4xlarge, m5n.24xlarge

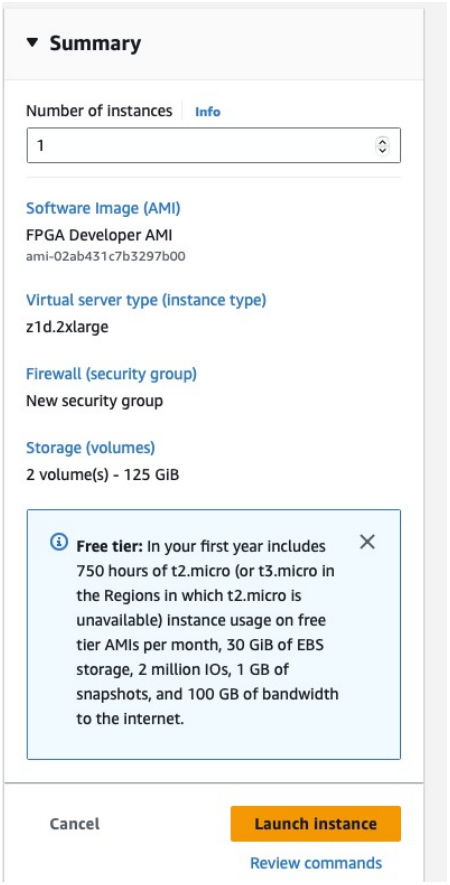
Key pair



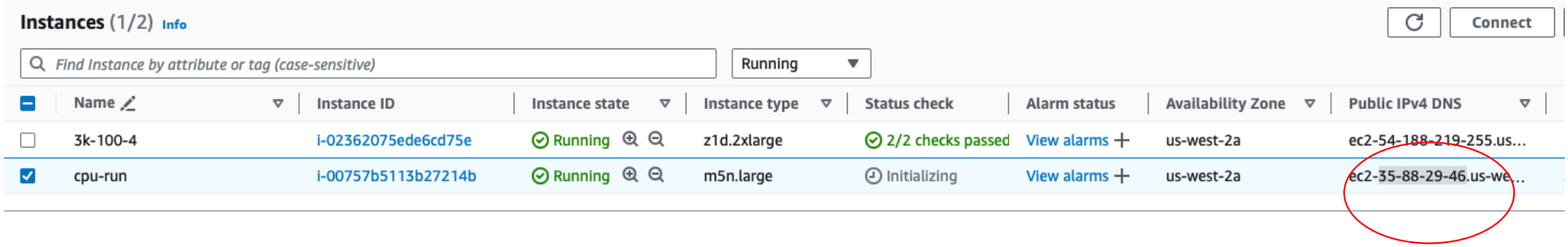
For information on how to create a new key pair go here

Launch m5n.large Instance

Repeat the same for all other CPU instances

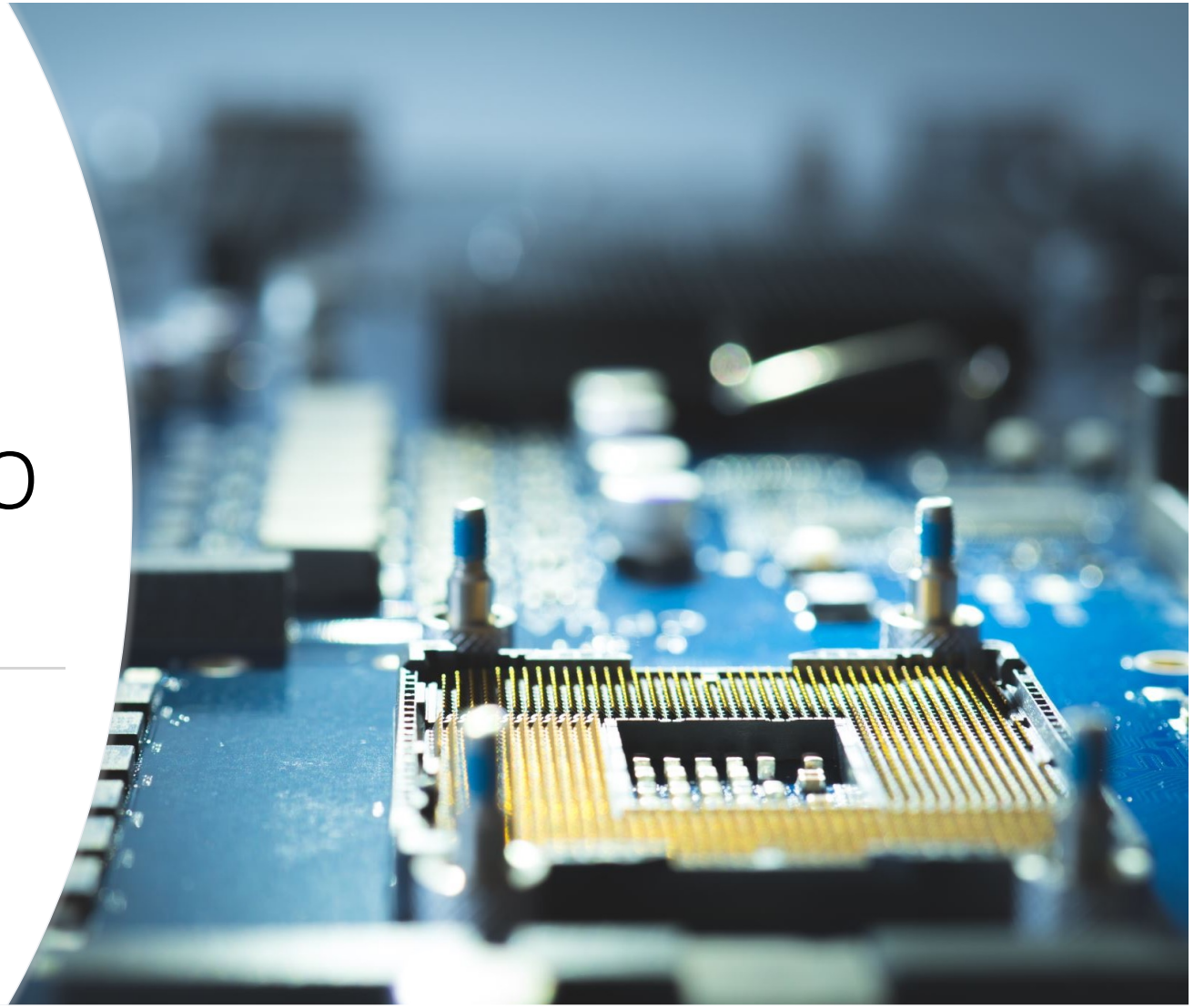


EC2 Instances



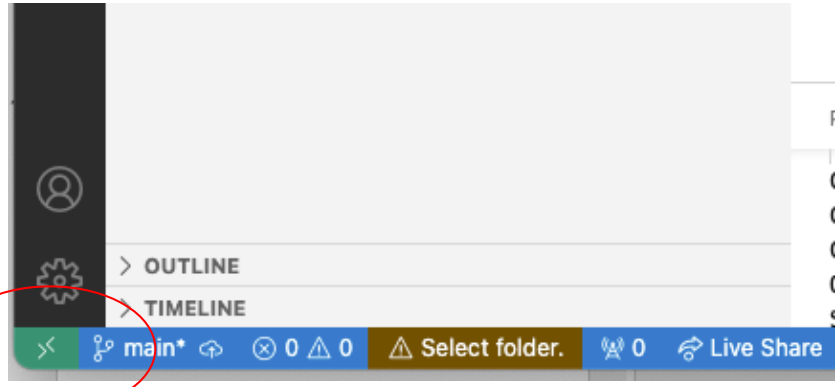
- In the top-left menu, select 'Instances’
- Copy the public IPv4 address in Visual Studio code

VISUAL STUDIO CODE



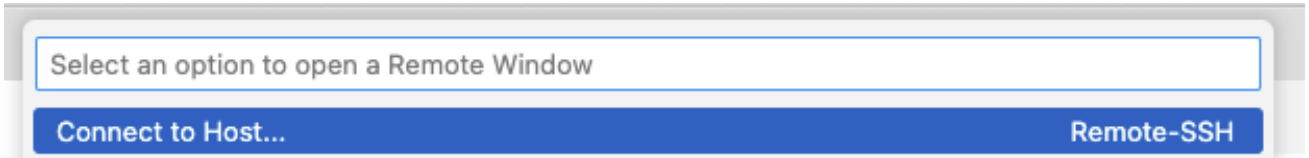
Open a Remote Window

- On the bottom-left corner of Visual Studio Code click on the green button ‘Open a Remote Window’
- Click on Connect to Host
- Click on Configure SSH Hosts.
  - Give a name to the Host (e.g. aws-ec-cpu)
  - Copy the public IP address (red circle) and set the path to your key
  - Set the user to **ec2-user**
- Connect to aws-ec2.




```
# ------------------------------ CONNECT TO AMAZON AWS
Host aws-ec-cpu
HostName ec2-34-219-191-114.us-west-2.compute.amazonaws.com
User ec2-user
IdentityFile ~/[YOUR KEY PATH].pem
# ----------------------------
```

CLICK the GREEN BUTTON HERE



Setup the Instance

Install utilities and Copy Repo

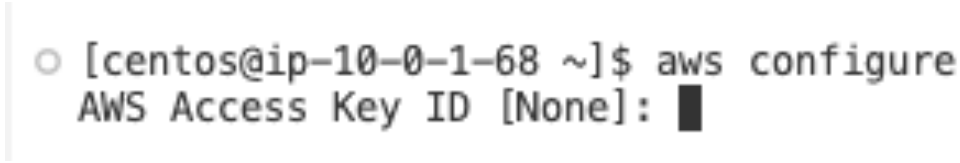
- Initiate a terminal session on the AWS instance and run the subsequent script to install the utilities

```
sudo yum install git -y
sudo yum install make -y
sudo yum install tmux -y
sudo yum install wget -y
```

- Clone our GitHub repository into a directory of your preference (e.g., /home/ec2-user):

```
git clone https://github.com/AleP83/FPGA-Econ.git
```

AWS Configure



1. Go to your aws account and set (one time thing):
   - AWS Access Key ID
   -AWS Secret Access Key
2. Go to the terminal in visual studio and type aws configure

```
aws configure
 AWS Access Key ID [**************xxxx]: <Your AWS Access Key ID>
 AWS Secret Access Key [**************xxxx]: <Your AWS Secret Access Key>
 Default region name: [us-west-1]: us-east-1
 Default output format [None]: json
```

3. Set:
   - AWS Access Key ID:
   - AWS Secret Access Key
   - Default Region name: us-west-2
     Note: this depends in which region you launched your instance.
   - Default output format [json]: json

Modify the Makefile

- Set the AWS S3 Bucket Name. Specify the S3 bucket name by replacing S3-NAME-GOES-HERE

```
S3_EXE_BUCKET_NAME := S3-NAME-GOES-HERE
```

Remark: The S3 bucket name must be globally unique within AWS. If an error occurs during bucket creation, it may be due to the name being already in use by another user.

- Select the AWS region of the S3 bucket (default is us-west-2):

```
AWS_REGION := us-west-2
```

m5n.large

Run all binaries in m5n.large

- Log into an m5n.large instance, previously set up following the instructions above
- **Initiate `tmux` terminal session**. To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single m5n.large instance

```
make cpu_results M5N=1x USE_AWS_S3_EXE=yes
```

Faster Replication Strategy.

- Set up three separate m5n.large instances as described above.
- Run the following scripts from the terminal of each instance:

```
make cpu_results M5N=1xBATCH1 USE_AWS_S3_EXE=yes
```

```
make cpu_results M5N=1xBATCH2 USE_AWS_S3_EXE=yes
```

```
make cpu_results M5N=1xBATCH3 USE_AWS_S3_EXE=yes
```

m5n.4xlarge

Run all binaries in m5n.4xlarge

- Log into an m5n.4xlarge instance, previously set up following the instructions above
- **Initiate `tmux` terminal session**. To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single m5n.4xlarge instance

```
make cpu_results M5N=4x USE_AWS_S3_EXE=yes
```

m5n.24xlarge

Run all binaries in m5n.24xlarge

- Log into an m5n.24xlarge instance, previously set up following the instructions above
- **Initiate `tmux` terminal session**. To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single m5n.24xlarge instance

```
make cpu_results M5N=24x USE_AWS_S3_EXE=yes
```