Launch an Ubuntu OS based Xilinx AMI:

- Generate a new key pair if it is the first time you are creating an instance, otherwise you won't be able to connect to it.
- This will be the only chance to get the private key file, so you need to save the private key file in a safe location (e.g. ~/.ssh/<keyname>.pem on Linux or C:\Users\<username>\.ssh\keyname>.pem)
- Once the instance is run, get the Public DNS which you will need to connect to the Xilinx AMI instance.
- Links for getting started:
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/launching-instance.html
  - https://aws.amazon.com/getting-started/tutorials/launch-a-virtual-machine/

Connect to Xilinx AMI without GUI:

- **Connection through SSH:**
  - **NOTE:** Contact your sys admin before trying to connect through SSH to provide you with the correct command in case you need proxy setup.
  - Change the permissions of the private key file
    - $ chmod 400 <keyname>.pem
  - $ ssh -i <keyname>.pem ubuntu@<public_DNS_name>
    - The public DNS for your instance can be retrieved using the Amazon EC2 console. if Public DNS column is hidden, unhide using Show/Hide icon.
- Connect to Amazon EC2 Instance:
  - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html#AccessingInstancesLinuxSSHClient

Connect to Xilinx AMI Instance with GUI:

- **Pre-requisite:**
  - Install and configure the desktop environment by connecting to instance through SSH as described above:
    - $ sudo apt update
    - $ sudo apt install ubuntu-desktop
- **Connection via VNC:**
  - **NOTE:** Contact your sys admin before trying to connect through SSH to provide you with the correct command in case you need proxy setup.
  - Change permissions of the private key file
    - $ chmod 400 <keyname>.pem
  - $ ssh -L <HOSTNAME>:59XX:localhost:59XX -i <keyname>.pem ubuntu@<public_DNS_name>

        - where HOSTNAME is the same host where you are running ssh from.
        - where XX is the VNC port display to be used (e.g. 01)
    - Install the pre-requisite desktop environment if not already installed
    - $ sudo apt install tigervnc-standalone-server tigervnc-common tigervnc-xorg-extension
    - $ vncserver :XX
        - Set a password when prompted
    - $ sudo passwd ubuntu
        - Set a password when prompted
    - Connect to the VNC session from the same host where you connected to the Xilinx AMI using ssh.
        - $ vncclient <HOSTNAME>:XX
- **Connection via RDP:**
    - **NOTE:** Contact your sys admin before trying to connect through SSH to provide you with the correct command in case you need proxy setup.
    - Change permissions of the private key file
        - $ chmod 400 <keyname>.pem
    - $ ssh -L <HOSTNAME>:3389:localhost:3389 -i <keyname>.pem ubuntu@<public_DNS_name>
        - where HOSTNAME is the same host where you are running ssh from.
    - Install the pre-requisite desktop environment if not already installed
    - $ sudo apt install xrdp
    - $ sudo systemctl enable xrdp
    - $ sudo systemctl start xrdp
    - $ sudo passwd ubuntu
    - Set a password when prompted
    - Connect to the RDP session from the same host where you connected to the Xilinx AMI using ssh
        - <HOSTNAME>:3389

Copy files to Xilinx AMI:

- **NOTE:** Contact your sys admin before trying to connect through SSH to provide you with the correct command in case you need proxy setup.
- $ scp -i <pem-name>.pem <source_file_from_HOSTNAME> ubuntu@<public_DNS_name>:/home/ubuntu/<directory>

Copy files from Xilinx AMI:

- **NOTE:** Contact your sys admin before trying to connect through SSH to provide you with the correct command in case you need proxy setup.
- $ scp -i <pem-name>.pem ubuntu@<public_DNS_name>:<file_path_to_copy_from> <output_path_from_HOSTNAME>

Running Example Design with Vivado

The following steps will ensure that your instance is functioning correctly.

1. Please open a terminal and type “vivado” into the terminal to open Vivado IDE with GUI.
    a. For additional quick reference commands (such as batch / TCL shell mode) please refer to our 1-page guide: https://docs.xilinx.com/v/u/2018.2-English/ug975-vivado-quick-reference
2. In the green “Quick Start” section, please select “Open Example Project”.
3. Click “Next” on the dialog box
4. Select “Base Microblaze” from the templates and click “Next”.
5. Create a project name and select your chosen directory. Then, click “Next”.
6. Choose a given board eval platform, and click “Next”. Then, select “Finish” in the last dialog box.
7. Once your project is finished initializing, please generate a bitstream by selecting the following icon at the top toolbar: 
8. Vivado will ask if you would like to first generate synthesis and implementation. Select “Yes”.
9. Choose a launch directory in the dialog box and select “Ok”.
10. Vivado will tell you when your run is successfully completed. It will then tell you when you have generated a bitstream via a new dialog box.
    a. You have now successfully generated a bitstream for this project. Please refer to your TCL Console to find where your run output is stored.
    b. Copy the bitstream as needed to your host machine
11. For an additional tutorial, please refer to our IP Integrator lab: https://docs.xilinx.com/r/en-US/ug995-vivado-ip-subsystems-tutorial