Practical Tasks SSM Run Command and Automation

**Task1 - Create Systems Manager IAM role**

1. go to **Identity Access Management (IAM)** and click on **Roles**.
2. Click on **Create Role**, and choose the service that will use this role, it should be **EC2**. Click **Next**.
3. Attach a predefined policy **AmazonEC2RoleforSSM** directly to this role and click on **Next**
4. Create Tag **Name**= **MySystemsManagerRole** and click on **Next**
5. Give a name to the role, e.g. **MySystemsManagerRole** and click on **Create Role**

**Task2 – Create EC2 instance(s) with SSM IAM role**

1. Back to **EC2** console and click on **Launch Instance**.
2. Select the **Amazon Linux 2 AMI**
3. Select the **t2.micro** and click on **Next**
4. Change **IAM role** only and select **MySystemsManagerRole (**created in Task 1**)** in the list
5. Click on **Next** two times
6. Add **Tags, Name=MyTest1,** **Application**=**TestApp**. It will be used for identifying our system in Systems Manager. Click **Next**
7. Click **Review** and then **Launch**
8. Select the **default** key pair as we don’t plan to log in to the instance by SSH and click on **Launch Instance**
9. **Click**
10. If you want to create additional instances just repeat 5-11 steps and change Tag **Name = MyTest2**, **Name=MyTest3** and etc

**Task 3 – Systems Manager Run Command**

1. Open **Systems manager** and click on **Automation** in **Node Management** section
2. Click on **Run Command** in the right panel
3. In the Search field insert **AWS-RunShell** and press **Enter** key
4. Select **AWS-RunShellScript** and this allows you to either specify a shell script or you can just type the command that you want to run.
5. Let’s type **ifconfig -a** in **Command parameter** section
6. And in the **Targets** section select **Choose Instances manually**
7. In the **Instances Search** field apply a filter **Ping status**=**Online**
8. Select your instances and click on **Run**
9. Click on **Refresh** to see the status and it should be successful.

Let’s take a look at the output of that command click on one of the instances and click on **Output**. There is the output of the **ifconfig -a** command that we've just run.

**Task 4 – Systems Manager Automation**

1. Open **Systems manager** and click on **Automation** in **Change management** section
2. click on **Execute Automation** in the right panel
3. In the **Search** field insert **AWS-StopEC2Instance** and press **Enter** key
4. Select **AWS-StopEC2Instance** automation document and click on **Next**
5. In the **Input parameters section** Enable **Show Interactive Instances Picker**
6. Select instances that you created in the **Task2** and click on **Execute**.
7. If task is successful that will stop those EC2 instances automatically. It can sometimes take a few minutes to complete executing. But if it has been successful, you should see the green tick and success.
8. Check it by going to **Services -> EC2** management console.
9. Click on **Instances** and you should see that the instance or instances that you chose have stopped.
10. If you don’t need to have those EC2 instances any more select them and in the Instance state drop down menu select click on Terminate instance and click on Terminate in the pop-up window