Candidate ID: **7032164**

Candidate Name: **Pavithra Ravi**
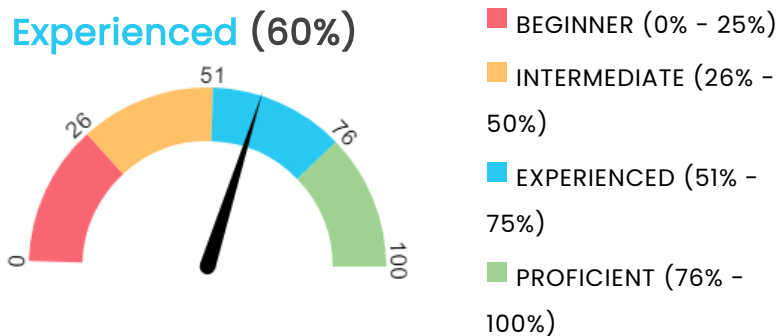
Date: **30-Aug-2022**

Assessment Name:
 LatentView Analytics- DevOps Training

Client Name: **Simplilearn**

SCORE ANALYSIS

Score: **27/ 45**

Time Taken:
 84 min 37 sec / 90 min

Experienced (60%)

- BEGINNER (0% - 25%)
- INTERMEDIATE (26% - 50%)
- EXPERIENCED (51% - 75%)
- PROFICIENT (76% - 100%)

Pavithra Ravi scored **60%** and completed assessment in **94%** of the alloted time

SECTION SCORE ANALYSIS

- BEGINNER (0% - 25%)
- INTERMEDIATE (26% - 50%)
- EXPERIENCED (51% - 75%)
- PROFICIENT (76% - 100%)

Section	Percentage
DevOps Essentials	13/18 (72%)
DevOps - Subjective	6/15 (40%)
Jenkins & Docker	8/12 (67%)

SECTION SKILL ANALYSIS

Section 1: DevOps Essentials

Total Score: **13/ 18**   Negative Points: **0**   Time Taken: **30 min 36 sec/36 min**

Question Analysis:

Total Question: **18**   Correct: **13**   Wrong: **5**   Skipped: **0**   Not Answered: **0**

Skills	#Questions	Skill Score
Version Control	6	3/6
DevOps Essentials	5	5/5
Git	7	5/7

Section 2: DevOps - Subjective

Total Score: **6/ 15**   Negative Points: **0**   Time Taken: **30 min /30 min**

Question Analysis:

Total Question: **3**   Correct: **2**   Wrong: **1**   Skipped: **0**   Not Answered: **0**

Skills	#Questions	Skill Score
Docker AI-Logicbox	1	0/5
Jenkins AI-LogicBox	1	2/5
Git AI-LogicBox	1	4/5

Section 3: Jenkins & Docker

Total Score: **8/ 12**   Negative Points: **0**   Time Taken: **23 min 59 sec/24 min**

Question Analysis:

Total Question: **12**   Correct: **8**   Wrong: **4**   Skipped: **0**   Not Answered: **0**

Skills	#Questions	Skill Score
Jenkins CI/CD Integration	4	3/4
Docker	8	5/8

DIFFICULTY LEVEL ANALYSIS

Level	Number of Questions	Correct Attempts	Correctness
Easy	18	15	83.33%

imocha

Level	Number of Questions	Correct Attempts	Correctness
Medium	6	3	50%
Hard	9	5	55.56%

PROCTORING ANALYSIS

Images Captured: 176   **Image Violations: 2**

🚩 Image violations detected, within tolerable limit.

Window Violation: 76   **Time Violation: 33 min 50 sec**

🔖 Window violations detected above 1 min, beyond tolerable limit.

TEST LOG

No data available

QUESTION DETAILS

Question: **#1**	Type: **AI-LogicBox**	Skill: **Git AI-LogicBox**	Status: **Answered**
Result: **Correct**	Level: **Easy**	Time Taken: **8 min 32 sec**	Average Time: **10 min 38 sec**
Score: **4 / 5**	Window Violation: **3 times**	Time Violation: **2 min 18 sec**	

Question #1

You have a new feature that you need to add to an application without any effect. The rest of the team should work normally without any delay. You want to write the story with the Git command to perform the given task.

A few lines in the Sample Script are missing (Enter your code here). You need to complete the code as per the given instructions.

- Create a new branch and name it new-feature.

```
Blank 1: Enter your code here
```

- Switch the working to the new branch you just created.

imocha

```
Blank 2: Enter your code here
```

- Check to pull the latest version of the code.

```
Blank 3: Enter your code here
```

- Add some code for the new feature, then add these changes to the new branch git add. Commit the changes you created with a suitable message.

```
Blank 4: Enter your code here
```

- Push the new change to the branch you created.

```
Blank 5: Enter your code here
```

Answer:

Answer1   ✓ **Correct**   **Points:** 1   **Score:** 1

git branch new-feature

Answer2   ✓ **Correct**   **Points:** 1   **Score:** 1

git checkout new-feature

Answer3   ✗ **Wrong**   **Points:** 1   **Score:** 0

git pull <remote-repo-url>

Answer4   ✓ **Correct**   **Points:** 1   **Score:** 1

git commit -a -m "adding changes"

Answer5   ✓ **Correct**   **Points:** 1   **Score:** 1

git push origin new-feature

Question: **#2**	Type: **AI-LogicBox**	Skill: **Docker AI-Logicbox**	Status: **Answered**
Result: **Wrong**	Level: **Easy**	Time Taken: **13 min 2 sec**	Average Time: **10 min 32 sec**
Score: **0 / 5**		Window Violation: **9 times** Time Violation: **7 min 16 sec**	

Question #2

In Docker, an image is run in a Docker container. Containers are instances of Docker images that can be run using the Docker run command. You are working on a Docker image called hello-world that is deployed using a Docker container.

In relation to the Docker container specifications, complete the code as per the given instructions.

Write the Bash command to find the Docker container ID through which the app "hello-world" has

been deployed.

OUTPUT = $(`Blank 1: Write your code here`)

Write the command to list the top processes of the container associated with hello-world. Use the container ID variable obtained in blank 1.

`Blank 2: Write your code here`

Write the command to list the container statistics such as the memory usage, CPU usage, etc. of the container associated with hello-world. Use the container ID variable obtained in blank 1.

`Blank 3: Write your code here`

Write the command to stop the container associated with hello-world. Use the container ID variable obtained in blank 1.

`Blank 4: Write your code here`

Write the command to remove the container associated with hello-world. The container can be in any state when the remove command is run. Use the container ID variable obtained in blank 1.

`Blank 5: Write your code here`

Answer:

Answer1   ✗ **Wrong**   **Points:** 1   **Score:** 0

```
docker ps -aqf "name=hello-world"
```

Answer2   ✗ **Wrong**   **Points:** 1   **Score:** 0

```
docker top OUTPUT
```

Answer3   ✗ **Wrong**   **Points:** 1   **Score:** 0

```
docker stats OUTPUT
```

Answer4   ✗ **Wrong**   **Points:** 1   **Score:** 0

```
docker stop OUTPUT
```

Answer5   ✗ **Wrong**   **Points:** 1   **Score:** 0

```
docker rm -f OUTPUT
```

Question: **#3**	Type: **AI-LogicBox**	Skill: **Jenkins AI-LogicBox**	Status: **Answered**
Result: **Correct**	Level: **Easy**	Time Taken: **8 min 28 sec**	Average Time: **6 min 56 sec**
Score: **2 / 5**	Window Violation: **5 times**	Time Violation: **2 min 12 sec**	

Question #3

You want to send a good morning message every day at 9:00 AM to an employee's email. You want to do this with Jenkins using triggers and schedules and then send the message via email. You need to save your message on a parameter and the type should be a string.

Complete the code as per the given instructions:

At Blank 1: Define the String variable and name it MESSAGE_TEXT.

At Blank 2: Create a suitable schedule to run every day at 9:00 AM.

At Blank 3: Use the environment that you created on the echo command.

At Blank 4: In case of success, send an email to me@example.com with SUCCESS in the subject and use the variable in the body.

At Blank 5: Use the right code in case of failure.

A few lines in the Sample Script are missing (Enter your code here). You need to complete the code as per the given instructions.

Sample Script:

```
pipeline {

    agent any
    environment {
```

```
Blank 1: Enter your code here
```

```
    }
    triggers {
```

```
Blank 2: Enter your code here
```

```
    }
    stages {
        stage("good morning") {
            steps {
```

```
Blank 3: Enter your code here
```

```
            }
        }
    }
    post {
        success {
            sh "echo 'Send mail on success'"
            mail to:
```

```
Blank 4: Enter your code here
```

```
        }
```

```
Blank 5: Enter your code here
```

```
    {
            sh "echo 'Send mail on failure'"
            // mail to:"me@example.com",
subject:"FAILURE:
${currentBuild.fullDisplayName}", body:
```

```
                                    "Ohhh, it is failed"
                                          }

                                    }
                              }
```

Answer:

Answer1   ✓ Correct   **Points:** 1   **Score:** 1

MESSAGE_TEXT : "Good morning"

Answer2   ✗ Wrong   **Points:** 1   **Score:** 0

schedule : * 9 * * *

Answer3   ✓ Correct   **Points:** 1   **Score:** 1

sh "echo $MESSAGE_TEXT'"