**EXAM**

|  |
| --- |
| There are four sections in the assessment: various theory questions, a task to describe processes/definitions/structures or operations depicted in a diagram, a practical exercise to host a very small static web site on AWS and, finally, a task to create a new Docker Image and host this containerized application on AWS. You are welcome to use personal notes made throughout the course sessions to complete assessment tasks.If you use any external resources, please make sure to properly reference them in your answers.  You are given 2 hours to complete this assessment. We recommend that the best way to manage your time is to allocate approximately 30 min for each section and then move on to the next task. The maximum possible score that can be awarded for this test is 100 points, which is equivalent to 100% grade. This test contributes 90% weight to the overall course grade.  Good luck! |

|  |  |  |
| --- | --- | --- |
|  | **SECTION** | **POINTS** |
| **A** | Theory questions | **30** |
| **B** | Diagram description | **20** |
| **C** | Static web-site with AWS | **20** |
| **D** | Docker Image with AWS | **30** |
|  | **TOTAL** | **100** |

**Theory Questions (30 points total)**

**The following 10 questions are multiple-choice. Please choose one correct answer unless specified otherwise. You get awarded 1 point for each correctly answered question.**

**Q1)** Which managed service could provide distribution of incoming traffic across multiple Amazon EC2 instances?

A. Amazon Elasticsearch

B. AWS Elastic Beanstalk

**C. Elastic Load Balancing**

D. AWS Auto Scaling

**Q2)** Which element of AWS Global infrastructure consists of multiple Availability Zones?

A. Edge Location

**B. Region**

C. Point of Presence

D. Data Centre

**Q3)** What feature of a Virtual Private Cloud (VPC) works like a virtual firewall for the network?

A. Subnets

B. Virtual Private Gateways

**C. Security Groups**

D. Internet Gateways

**Q4)** You are looking to launch a static website with minimal configuration. Which of the following AWS services provides this feature?

A. Amazon CloudWatch

B. Amazon Redshift

C. AWS CodeDeploy

**D. Amazon S3**

**Q5)** Which online tool provides guidance in following AWS best practices by providing personalised recommendations for your cloud workloads?

1. **AWS Trusted Advisor**
2. AWS Personal Health Dashboard
3. AWS Well-Architected Framework
4. Amazon CloudWatch

**Q6)** What is the minimum AWS support plan that provides the complete set of AWS Trusted Advisor checks?

1. Developer
2. **Business**
3. Enterprise
4. Basic

**Q7)** Your US-based team wants to expand an internal application to your office in Singapore. They are able to take the application and spin up a version in the ap-southeast-1 region. Which benefit of cloud computing is best illustrated with this scenario?

1. Stop guessing capacity
2. **Go global in minutes**
3. Stop spending money running and maintaining data centres
4. Trade capital expense for variable expense

**Q8)** Your organisation is looking to test a new web application concept. The team set up an AWS account and infrastructure within a day on which they will build a prototype for this application. Which cloud computing benefit does this illustrate?

1. Go global in minutes
2. Stop guessing capacity
3. **Increase speed and agility**
4. Hybrid cloud

**Q9)** Which element of the AWS Global Infrastructure is made up of one or more data centres connected by a low latency network with redundant power, networking, and connectivity within an AWS Region?

1. Edge Location
2. Regional Edge Cache
3. Point of Presence
4. **Availability Zone**

**Q10)** You want to have email support within business hours from AWS for an application you are building. What is the minimum support plan that provides this?

1. Basic
2. Enterprise
3. **Business**
4. Developer

**The following 10 questions are also multiple-choice and they are related to Docker. You will be awarded 1 point for each correctly answered question.**

**Q11)** Which of these is NOT true about containers?

**A. Containers include the operating system as well as code and libraries**

B. Containers are smaller and more portable than virtual machines

C. Multiple containers can run on a single physical server

D. Containers allow easy scaling of services

**Q12)** What is a Dockerfile?

A. A developer who loves Docker and containerization, frequently espousing its virtues

B. Any folder or document to run Docker components

**C. A template used to describe the build of an image**

D. Any report or document that Docker components produce

**Q13)** What command do we use to completely get rid of a Docker Image?

A. docker delete

**B. docker rmi**

C. docker stop

D. docker kill

**Q14)** How do you manage Docker images?

**A. Docker Hub**

**B. Docker Registry**

C. Docker Swarm

D. Both A & B

**Q15)** What is Docker?

**A. an open-source lightweight containerisation technology**

B. an operating system

C. a language

D. None of the above

**Q16)** What are the important features of Docker?

A. Easy configuration

B. Version control

C. Management of container scaling

**D. All of the above**

**Q17)** What is a Docker image?

**A. a read-only set of instructions for creating a container**

B. a graphical user interface for creating containers

C. a language

D. None of the above

**Q18)** What is Docker Engine?

**A. an open-source containerisation technology for building and containerising your applications**

B. an operating system

C. a language

D. a graphical user interface for creating containers

**Q19)** What is Docker Swarm?

**A. a container orchestration tool**

B.  a read-only set of instructions for creating a container

C. a centralisation platform

D. None of the above

**Q20)** What is Docker Hub?

**A. a service provided by Docker for finding and sharing container images with your team**

B. a centralised tool

C. a container run time

D. All of the above

**The following 10 questions are also multiple-choice and they are related to Kubernetes. You will be awarded 1 point for each correctly answered question.**

**Q21)** What is the basic operational unit of Kubernetes?

**A. Pod**

B. Task

C. Nodes

D. None of the above

**Q22)** What are the main components of Kubernetes?

A. Scheduler

B. API server

C. Controller manager

**D. All of the Above**

**Q23)** Which of the following runs on each node and ensures containers are running in a pod?

A. Pod

B. Etcd

**C. Kubelet**

D. All of the Above

**Q24**) Who developed Kubernetes?

**A. Google**

B. Apple Inc.

C. IBM

D. Microsoft

**Q25**) Kubernetes is a <…> platform.

A. Portable

B. Extensible

C. Open-source

**D. All of the Above**

**Q26)** A node is a <…> in Kubernetes.

A. developing tool

**B. worker machine**

C. local machine

D. None of the above

**Q27**) Which of the following manages assigning nodes to pods depending on resource availability?

A. Etcd

B. Kubectl

**C. Scheduler**

D. None of the above

**Q28**) Kubernetes is written in which programming language?

**A. Go**

B. Html

C. Php

D. Python

**Q29**) To create a new deployment in Kubernetes, which of the following commands should you use?

**A. kubectl run**

B. kubernetes get deployment

C. kubernetes set deployment

D. None of the above

**Q30**) Which command is used to create a kubernetes service.

A. kubectl run

B. kubectl deploy

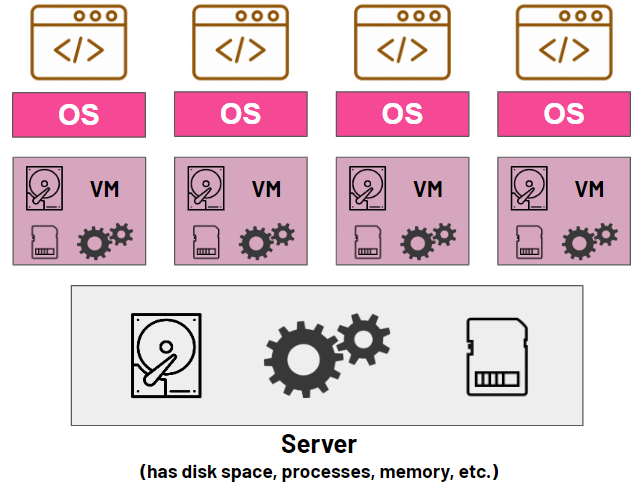
**C. kubectl expose**

D. docker service create

**Diagram Description (20 points total)**

**Given the diagrams below, please describe these two models in detail and compare how their efficiency may differ from one another. You will be awarded 10 points for the correct description of these models and further 10 points for providing the “key points” comparison analysis.**

**Model 1**



**An explanation of the Virtual Machines (VMs) models:**

**On top of a hypervisor that is deployed on a physical server, each virtual machine (VM) runs its own operating system (OS). On a single server, this hypervisor oversees several virtual machines.**

**Each VM contains its own OS, virtual hardware (such CPU, memory, and storage), and applications. Because of this full stack, virtual machines (VMs) are larger and use resources less effectively.**

**Each VM needs a complete operating system, VMs need more resources. Increased memory and storage overhead may result from this.**

**Conatiners: While application processes are isolated from one another and share the same OS kernel, containers do not share this feature. They contain the program and all of its dependencies and are lightweight.**

**Containers are more compact because they only contain the libraries and binaries required for the program to function. Containers use resources more effectively, enabling shorter startup times and improved scalability, because they share the host operating system.**

**Virtual machines (VMs): Greater overhead because each VM runs a whole OS. If not handled appropriately, this could result in the waste of resources.**

**Containers: Reduce resource consumption by enabling the operation of several containers on a single host without the need for separate operating systems.**

**Virtual machines (VMs): Strong isolation because every VM is totally independent and has its own OS. Performance may suffer as a result, but security may be strengthened.**

**Containers: Offer process-level isolation, which is not as reliable as virtual machine isolation. They do, however, make communication between apps easier and are enough for a large number of applications.**

**VMs: Longer starting times because the OS must be completely booted. Performance can be affected by virtualization overhead, particularly in I/O activities.**

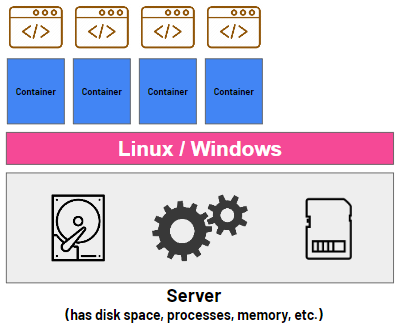
**Containers: Since they don't require OS booting, they start almost instantaneously. Faster deployment and scaling are the outcome of this.**

**VMs: Because of their size and resource needs, they require more sophisticated management tools and procedures.**

**Containers: Scaling and load balancing can be automated with orchestration technologies like Kubernetes, making them easier to manage and deploy.**

Reference: [Containers vs. virtual machines (VMs) | Google Cloud](https://cloud.google.com/discover/containers-vs-vms)

**Model 2**



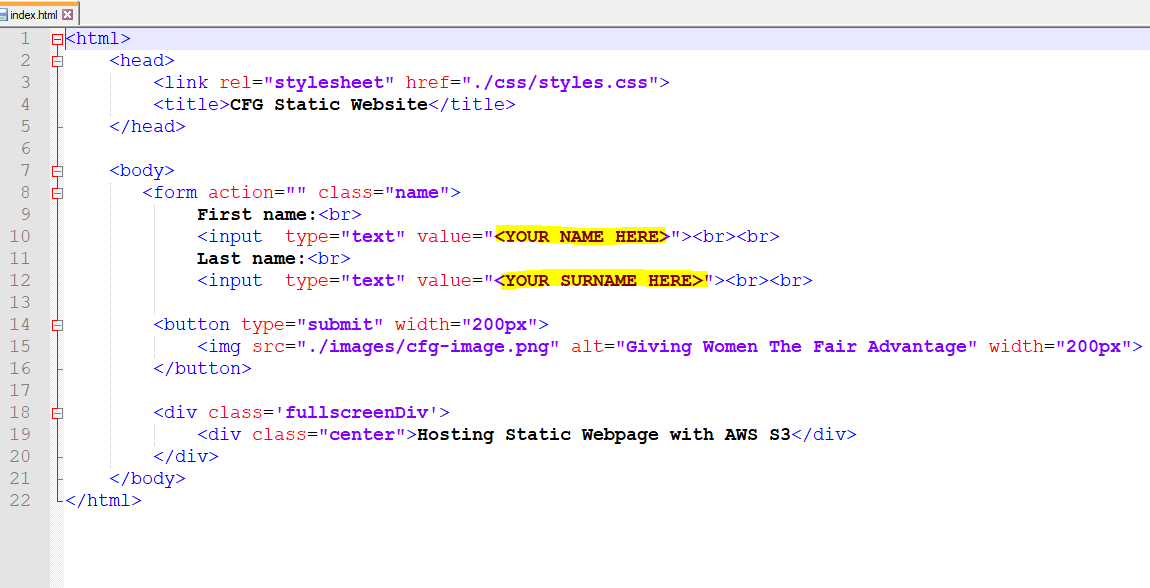
**Hosting a Static Website with S3 (20 points total)**

**Please use the contents of the folder called “static-webpage” that have been shared with you at the beginning of this assessment. Open index.html in any browser to see an example of a simple static webpage. The task is to register and host this webpage (and all required contents like images and css) on cloud with the help of AWS S3 service.**

[**CFG Static Website (cfg-master-exammn.s3-website-us-east-1.amazonaws.com)**](http://cfg-master-exammn.s3-website-us-east-1.amazonaws.com/static-webpage/index.html)

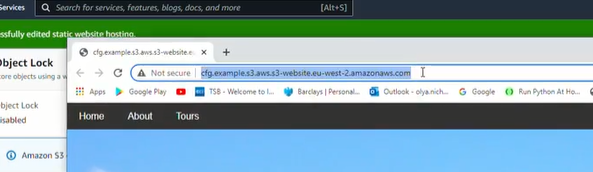
IMPORTANT NOTES:

1. Prior to posting your webpage online, please open **index.html** in any editor (even a notepad) and update YOUR name and surname as highlighted in yellow in a snippet below



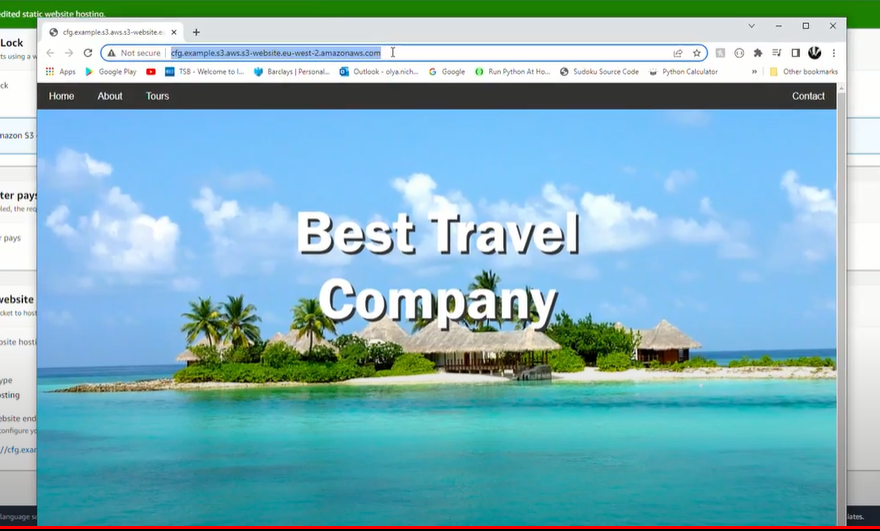
1. You need to submit a **URL address to your newly created website**, so that an instructor can access it from their machine when they mark this assessment.

Example URL from our previous lesson: (see the URL)



1. In order to be on a safe side, PLEASE TAKE A SCREENSHOT OF YOUR WEBSITE AND THE URL to prove that it is up and running, in case there are some technical issues in the future and an instructor is unable to access your link.

Example: (we need to clearly see your website and the URL)



1. **You need to keep your website LIVE for 5 days!** (Time required for instructors to mark all assessments). After one week you can take your website down (or sooner if your instructor says so). Please do not worry – it would NOT trigger any AWS payments as your free tier allows use of simple static webpages for thousands of hours.

**Hosting Docker App with AWS (30 points total)**

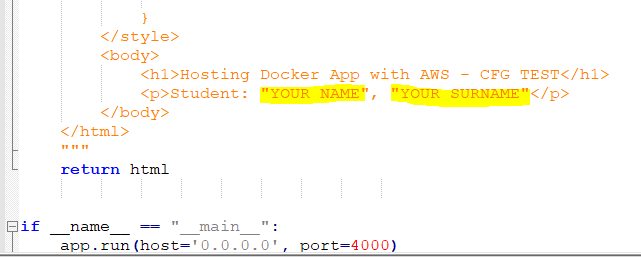
**Please use the contents of the folder called “docker-app” that have been shared with you at the beginning of this assessment. Open index.html in any browser to see an example of a simple static webpage. You DO NOT NEED the index.html script. It is only an example of what to expect**

**The task is:**

1. Use app.py and Dockerfile to create a new Docker Image
2. Add this new Docker App to AWS ECR
3. Then host the Docker App in the cloud with the help of AWS ECS
4. Your containerised App should be up and running on AWS and accessible by anyone who knows the URL.
5. **NB: if you are unable to configure and host your Docker App with AWS, then please do push your Docker Image to Docker Hub and SUBMIT the full Docker Image name (with your user name) instead, so that your instructor can see that you did build the Docker Image! You will be awarded 10 points for this.**

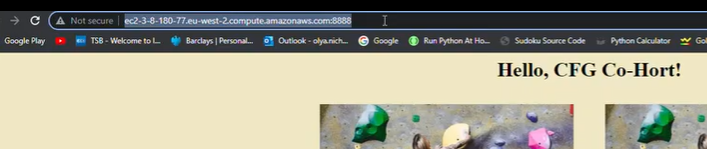
**IMPORTANT NOTES:**

1. Prior to posting your webpage online, please open **app.py** in any editor (even a notepad) and update YOUR name and surname as highlighted in yellow in a snippet below



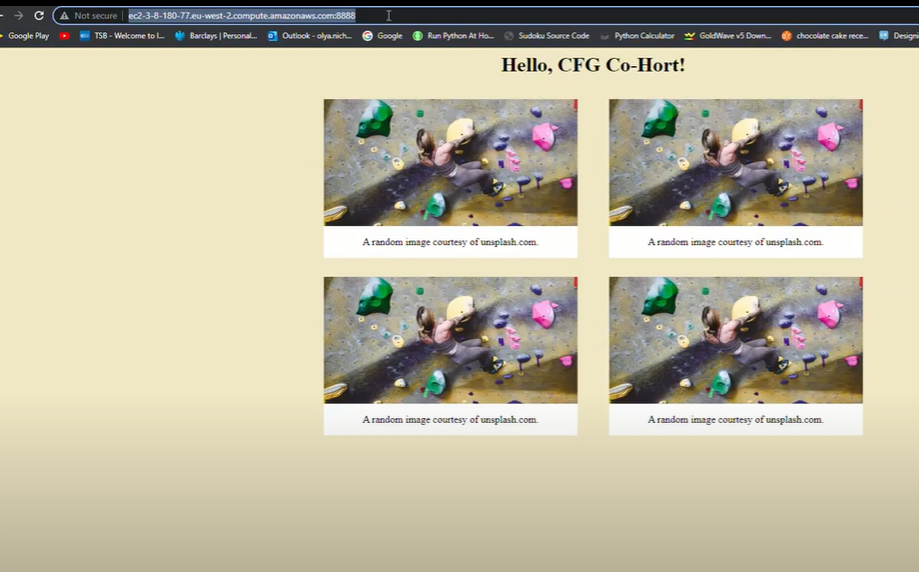
1. You need to submit a **URL address to your newly created website**, so that an instructor can access it from their machine when they mark this assessment.

Example URL from our previous lesson: (see the URL)



1. In order to be on a safe side, PLEASE TAKE A SCREENSHOT OF YOUR WEBSITE AND THE URL to prove that it is up and running, in case there are some technical issues in the future and an instructor is unable to access your link.

Example: (we need to clearly see your website and the URL)



1. **You need to keep your website LIVE for 5 days!** (Time required for instructors to mark all assessments). After one week you can take your website down (or sooner if your instructor says so). Please do not worry – it would NOT trigger any AWS payments as your free tier allows use of simple static webpages for thousands of hours.