1. 720 is the passmark out of 1000.

2. i.e the pass percentage is 72%. 3 out of 4 questions must be correct.

3. Total of 65 scored questions 47 questions must be answered corrctly.

4. 130 minutes to answer 65 questions. You have exactly 2 minutes to answer each question.

5. Topics tested are

DEPLOYMENT : 22% - 14 questions

[ Deploy written code in AWS using existing CI/CD pipelines, processes, and patterns

/ Elastic Bean Stalk

/ prepare the deployment package to be deployed to aws

/ deploy serverless applications]

SECURITY : 26% - 17 questions

[ Make authenticated calls to aws services

/ Implement encryption using aws services

/ Implement Application Authentication and Authurization]

DEVELOPMENT WITH AWS SERVICES : 30% - 19 questions

[ / Write code for serverless applications

/ Translate functional requirements in to application design

/ implement application design into application code

/ write code that interacts with aws services api's/ sdk's / aws cli

]

REFACTORING : 10% - 7 questions

[

/ optimize application to best use aws services and features.

/ Migrate existing application code to run on aws.

]

MONITORING AND TROUBLESHOOTING: 12% - 8 questions

[

/ write code that can be monitored

/ Perform root cause analysis on faults found in testing and production.

]

MOST IMPORTANT TOPICS ARE:

AWS LAMBDA :

|  |  |  |
| --- | --- | --- |
| Invocation Types | Using notifications and event source mapping | Concurrency and throttling |
| Xray + SQS + DLQ | Versions and Aliases | Blue green Deployment |
| Packaging and Deployment | VPC Connections with internet / NAT GW. | Lambda as ELB target. |
| Dependencies | Environment variables and encrypting them. |  |

DYNAMODB :

|  |  |
| --- | --- |
| Scan vs query [And the apis, parameters you can use] | Local and Global secondary index |
| Calculating Read and write Capacity units. | Performance and optimization best practices. |
| Use case : Session state , key value data store, scalability. | DynamoDB Streams |
| Use in serverless app with lambda and api gateway. | Dax – use cases |

API GATEWAY :

|  |  |
| --- | --- |
| Lambda /IAM / Cognito AUthorizers | Invalidation of cache. |
| Integration Types : Proxy vs Custom / aws vs http | caching |
| Import Export OpenAPI swagger specifications | Stage Variables |
| Performance Metrics. |  |

AWS COGNITO :

|  |  |  |
| --- | --- | --- |
| User pools vs Identity Pools | Unauthenticated identities | AWS cognito Sync |
| Using MFA with Cognito | Web Identity federation. |  |

AMAZON S3 :

|  |  |
| --- | --- |
| S3 : Encryption is very important for the exam | |
| S3 : Transfer Acceleration | Versioning |
| Copying data | Lifecycle Rules. |

IAM :

|  |  |
| --- | --- |
| IAM Policies and Roles | Cross account access |
| Multi-factor Authentication. | API calls |
| IAM roles with EC2 – Instance profiles | Access Keys vs Roles. |
| IAM best practices | Federation |

ECS :

|  |  |  |  |
| --- | --- | --- | --- |
| Shared storage between containers | | Single vs Multi Docker environments | |
| Uploading and downloading images with ECR | | Placement Stratagies. Spread, Binpack, random. | |
| Port Mappings | Defining Task Definitions | | IAM roles for Tasks. |

ELASTIC BEANSTALK :

AWS CLOUD FORMATION :

AMAZON CLOUD WATCH :

DEVELOPER TOOLS [CODE COMMIT / CODE BUILD / CODE DEPLOY / CODE PIPELINE / CODE STAR]

AMAZON CLOUD FRONT :

AWS XRAY :

AWS SQS :

AWS ELASTICACHE :

AMAZON STEP FUNCTIONS :

SSM PARAMETER STORE :

OTHERS : EC2 / LOAD BALANCING / AUTO SCALING / SNS / KMS / CLOUD TRAIL / AWS ORGANIZATIONS / AMAZON SIMPLE WORKFLOW SERVICE / AMAZON VIRTUAL PRIVATE CLOUD.  
  
Exam topics :  
https://digitalcloud.training/aws-developer-associate-exam/  
  
<https://medium.com/bb-tutorials-and-thoughts/how-to-pass-the-aws-certified-developer-associate-exam-fa9de075ec20>