Amazon

AWS-Certified-Big-Data-Specialty Exam

AWS Certified Big Data - Specialty

Questions & Answers
Demo

Practice Exam Questions - 2019

Question 1 **AWS Certified Big Data - Specialty**

Domain :Processing

Tick-Bank is a privately held Internet retailer of both physical and digital products founded in 2008. The company has more than six-million clients worldwide. Tick-Bank's technology aids in payments, tax calculations and a variety of customer service tasks and serve as a connection between digital content makers and affiliate dealers, who then promote them to clients thereby assist in building revenue making opportunities for companies.
Tick-Bank currently runs multiple java based web applications running on AWS and looking to enable web-site traffic analytics and also planning to extend the functionality for new web applications that are being launched. Tick-Bank uses KPL library to address event integration into the kinesis streams and thereby process the data to downstream applications for analytics. With growing applications and customers, performance issues are hindering real time analytics and need an administrator to standardize performance, monitoring, manage and costs by kinesis streams.
Please select 3 options.

- A. **Use multiple shards to integrate data from different applications, reshard by splitting hot shards to increase capacity of the stream**
- B. **Use multiple shards to integrate data from different applications, reshard by splitting cold shards to increase capacity of the stream**
- C. **Use CloudWatch metrics to monitor and determine the "hot" or "cold" shards and understand the usage capacity**
- D. **Use multiple shards to integrate data from different applications, reshard by merging cold shards to reduce cost of the stream**
- E. **Use multiple shards to integrate data from different applications, reshard by merging hot shards to reduce cost of the stream and improve performance**
- F. **Use CloudTrial metrics to monitor and determine the "hot" or "cold" shards and understand the usage capacity**

Explanation:

Answer: A, C, D

- **Option A is correct -** Splitting hot shards improve the performance. Define a single shard for each web application of a kinesis stream. Based on the usage generated through CloudWatch Metrics for each shard, split the hot shards or merge the cold shards. This way we can improve the

performance and reduce the costs for each stream. if the performance is still not addressed, adapt to different streams per application

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

- **Option B is incorrect -** Splitting cold shards does not improve the performance. Define a single shard for each web application of a kinesis stream. Based on the usage generated through CloudWatch Metrics for each shard, split the hot shards or merge the cold shards. This way we can improve the performance and reduce the costs for each stream. if the performance is still not addressed, adapt to different streams per application

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

- **Option C is correct -** CloudWatch Metrics determine which are your "hot" or "cold" shards, that is, shards that are receiving much more data, or much less data, than expected. You could then selectively split the hot shards to increase capacity for the hash keys that target those shards. Similarly, you could merge cold shards to make better use of their unused capacity.

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

- **Option D is correct -** Merging cold shards improve performance as well as costs. Define a single shard for each web application of a kinesis stream. Based on the usage generated through CloudWatch Metrics for each shard, split the hot shards or merge the cold shards. This way we can improve the performance and reduce the costs for each stream. if the performance is still not addressed, adapt to different streams per application

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

- **Option E is incorrect -** Merging hot shards does not improve performance. Define a single shard for each web application of a kinesis stream. Based on the usage generated through CloudWatch Metrics for each shard, split the hot shards or merge the cold shards. This way we can improve the performance and reduce the costs for each stream. if the performance is still not addressed, adapt to different streams per application

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

- **Option F is incorrect -** CloudTrial does not provide this. CloudWatch Metrics determine which are your "hot" or "cold" shards, that is, shards that are receiving much more data, or much less data, than expected. You could then selectively split the hot shards to increase capacity for the hash keys that target those shards. Similarly, you could merge cold shards to make better use of their unused capacity.

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding-strategies.html

Domain :Storage

Hymutabs Ltd (Hymutabs) is a global environmental solutions company running its operations in in Asia Pacific, the Middle East, Africa and the Americas. It maintains more than 10 exploration labs around the world, including a knowledge centre, an "innovative process development centre" in Singapore, a materials and membrane products development centre as well as advanced machining, prototyping and industrial design functions.
Hymutabs hosts their existing enterprise infrastructure on AWS and runs multiple applications to address the product life cycle management.

The datasets are available in Aurora, RDS and S3 in file format. Hymutabs Management team is interested in building analytics around product life cycle and advanced machining, prototyping and other functions.
The IT team proposed Redshift to fulfill the EDW and analytics requirements. They adapt modeling approaches laid by Bill Inmon and Kimball to efficiently design the solution. The team understands that the data loaded into Redshift would be in terabytes and identified multiple massive dimensions, facts, summaries of millions of records and are working on establishing the best practices to address the design concerns.
There are 6 tables that they are currently working on:

- ORDER_FCT is a Fact Table with billions of rows related to orders
- SALES_FCT is a Fact Table with billions of rows related to sales transactions. This table is specifically used to generate reports EOD (End of Day), EOW(End of Week), and EOM (End of Month) and also sales queries
- CUST_DIM is a Dimension table with billions of rows related to customers. It is a TYPE 2 Dimension table
- PART_DIM is a part dimension table with billions of records that defines the materials that were ordered
- DATE_DIM is a dimension table
- SUPPLIER_DIM holds the information about suppliers the Hymutabs work with

One of the key requirements includes ORDER_FCT and PART_DIM are joined together in most of order related queries. ORDER_FCT has many other dimensions to support analysis.
How would you design the distribution? Select 1 option.

A. **Distribute the ORDER_FCT with KEY distribution on its primary KEY (any one of the columns) and PART_DIM with KEY distribution on its PRIMARY KEY**

B. **Distribute the ORDER_FCT with ALL distribution on its primary KEY (any one of the columns) and PART_DIM with ALL distribution on its PRIMARY KEY**

C. **Distribute the ORDER_FCT with EVEN distribution on its primary KEY (any one of the columns) and PART_DIM with EVEN distribution on its PRIMARY KEY**

D. **Distribute the ORDER_FCT and PART_DIM on same key with KEY distribution**

E. **Distribute the ORDER_FCT and PART_DIM on same key with EVEN distribution**

Explanation:

Answer : D

- **Option A is incorrect -** KEY DISTRIBUTION distributes the rows are according to the values in one column. Queries initiate lot of redistribution of data of both ORDER_FCT and PART_DIM are not built on same key.

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option B is incorrect -** ALL distribution makes a copy of the entire table in every compute node. Being billion record tables, this is not a right approach to design.

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option C is incorrect -** EVEN DISTRIBUTION evenly distributes the rows across the slices in a round-robin fashion, regardless of the values in any particular column. EVEN distribution is appropriate when a table does not participate in joins. Definitely not a right approach.

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option D is correct -** KEY DISTRIBUTION distributes the rows are according to the values in one column. With distribution of data on same key in both the tables, there is no change of redistribution. This is the best approach to design.

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option E is incorrect -** EVEN DISTRIBUTION evenly distributes the rows across the slices in a round-robin fashion, regardless of the values in any

particular column. EVEN distribution is appropriate when a table does not participate in joins. Definitely not a right approach
https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

---

Question 3 **AWS Certified Big Data - Specialty**

Domain :Processing

MSP Bank, Limited is a leading varied Japanese monetary institution that provides a full range of financial products and services to both institutional and individual customers. It is headquartered in Tokyo. MSP Bank is hosting their existing infrastructure on AWS. MSP bank has many segments internally and they are planning to launch a self-data discovery platform running out of AWS on QuickSight.
Using QuickSight, multiple datasets are created and multiple analyses are generated respectively.

The Team is working on enabling auditing to track the records of actions taken by a user, role, or an AWS service in Amazon QuickSight. Also the team need to capture the logs and storage it for long term archival to address compliance. Please advice. Select 3 options.

- A. **Amazon QuickSight is integrated with AWS CloudTrail which provides a record of actions taken by a user, role, or an AWS service in Amazon QuickSight**
- B. **Amazon QuickSight is integrated with AWS CloudWatch which provides a record of actions taken by a user, role, or an AWS service in Amazon QuickSight**
- C. **when CloudTrail is enabled, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon QuickSight**
- D. **when CloudWatch is enabled, you can enable continuous delivery of CloudWatch events to an Amazon S3 bucket, including events for Amazon QuickSight**
- E. **If you don't configure a trail, you can still view the most recent events in the CloudTrail console in Event history**
- F. **If you don't configure a log, you can still view the most recent events in the CloudWatch console in Event history**

Explanation:

Answer: A,C,E
Amazon QuickSight is integrated with AWS CloudTrail. This service provides a record of actions taken by a user, role, or an AWS service in Amazon QuickSight. The calls captured include calls from the Amazon QuickSight console. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon QuickSight. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in Event history. Using the information collected by CloudTrail, you can determine the request that was made to Amazon QuickSight, the IP address from which the request was made, who made the request, when it was made, and additional details
https://docs.aws.amazon.com/quicksight/latest/user/logging-using-cloudtrail.html

Question 4 **AWS Certified Big Data - Specialty**

Domain :Storage

ConsumersHalt (CH) is an Indian department collection chain. There are 63 branches across 32 towns in India, with clothing, accessories, bags, shoes, jewelry, scents, faces, health and exquisiteness products, home furnishing and decor products.
CH runs their existing operations and analytics infrastructure out of AWS which includes S3, EC2, Auto Scaling, CDN and also Redshift. The Redshift platform is being used for advanced analytics, real time analytics and being actively used for past 2 years. Suddenly performance issues are occurring in the application and administrator being a superuser needs to provide a list of reports in terms of current

and historical performance of the cluster. What types of tables/views can help access the performance related info for diagnosis.
Select 3 options.

- [ ] A. **STL system tables are generated from Amazon Redshift log files to provide a history of the system. They serve logging.**
- [ ] B. **STL tables are actually virtual system tables that contain snapshots of the current system data. They serve snapshots.**
- [ ] C. **STV system tables are generated from Amazon Redshift log files to provide a history of the system. They serve logging.**
- [ ] D. **STV tables are actually virtual system tables that contain snapshots of the current system data. They serve snapshots.**
- [ ] E. **System views contain full data found in several of the STL and STV system tables.**
- [ ] F. **The system catalogs store schema metadata, such as information about tables and columns.**

Explanation:

Answer : A, D, F

- **Option A is correct -** STL system tables are generated from Amazon Redshift log files to provide a history of the system.

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_STL_tables.html

- **Option B is incorrect -** STL system tables are generated from Amazon Redshift log files to provide a history of the system.

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_STL_tables.html

- **Option C is incorrect -** STV tables are actually virtual system tables that contain snapshots of the current system data.

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_STV_tables.html

- **Option D is correct -** STV tables are actually virtual system tables that contain snapshots of the current system data.

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_STV_tables.html

- **Option E is incorrect -** System tables contain only subset of data

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_system_views.html

- **Option F is correct -** The system catalogs store schema metadata, such as information about tables and columns. System catalog tables have a PG prefix.

https://docs.aws.amazon.com/redshift/latest/dg/c_intro_catalog_views.html

---

Question 5 **AWS Certified Big Data - Specialty**

Domain :Processing

HikeHills.com (HH) is an online specialty retailer that sells clothing and outdoor refreshment gear for trekking, go camping, boulevard biking, mountain biking, rock hiking, ice mountaineering, skiing, avalanche protection, snowboarding, fly fishing, kayaking, rafting, road and trace running, and many more.
HH runs their entire online infrastructure on multiple java based web applications and other web framework applications running on AWS. The HH is capturing clickstream data and use custom-build recommendation engine to recommend products which eventually improve sales, understand customer preferences and already using AWS Kinesis Streams (KDS) to collect events and transaction logs and process the stream. Multiple departments from HH use different streams to address real-time integration and induce analytics into their applications and uses Kinesis as the backbone of real-time data integration across the enterprise.
HH uses a VPC to host all their applications and is looking at integration of kinesis into their web application. To understand the network flow behavior based on every 15 minutes, HH is looking at aggregating data based on the VPC logs for analytics. VPC Flow Logs have a capture window of approximately 10 minutes. What kind of queries can be used to capture aggregates based on each client for every 15 mins using Amazon Kinesis Data Analytics. Select 1 option.

A. **Stagger Windows queries**

B. **Tumbling Windows queries**

C. **Sliding windows queries**

D. **Continuous queries**

Explanation:

Answer: A

- **Option A is correct -** Stagger windows query, A query that aggregates data using keyed time-based windows that open as data arrives. The keys allow for multiple overlapping windows. This is the recommended way to aggregate data using time-based windows.

VPC Flow Logs have a capture window of approximately 10 minutes. But they can have a capture window of up to 15 minutes if you're aggregating data on the client. Stagger windows are ideal for aggregating these logs for analysis.
https://docs.aws.amazon.com/kinesisanalytics/latest/dev/stagger-window-concepts.html

- **Option B is incorrect -** Tumbling Windows query, A query that aggregates data using distinct time-based windows that open and close at regular intervals.

https://docs.aws.amazon.com/kinesisanalytics/latest/dev/tumbling-window-concepts.html

- **Option C is incorrect -** Sliding windows query, A query that aggregates data continuously, using a fixed time or rowcount interval.

https://docs.aws.amazon.com/kinesisanalytics/latest/dev/sliding-window-concepts.html

- **Option D is incorrect -** Continuous Query is a query over a stream executes continuously over streaming data. This continuous execution enables scenarios, such as the ability for applications to continuously query a stream and generate alerts.

https://docs.aws.amazon.com/kinesisanalytics/latest/dev/continuous-queries-concepts.html

---

Question 6 **AWS Certified Big Data - Specialty**

Domain :Processing

HikeHills.com (HH) is an online specialty retailer that sells clothing and outdoor refreshment gear for trekking, go camping, boulevard biking, mountain biking, rock hiking, ice mountaineering, skiing, avalanche protection, snowboarding, fly fishing, kayaking, rafting, road and trace running, and many more.
HH runs their entire online infrastructure on java based web applications running on AWS. The HH is capturing click stream data and use custom-build recommendation engine to recommend products which eventually improve sales, understand customer preferences and already using AWS kinesis KPL to collect events and transaction logs and process the stream. The event/log size is around 12 bytes. The log stream generated into the stream is used for multiple purposes. HH proposes Kinesis Firehose to process the stream and capture information. What purposes can be fulfilled OOTB without writing applications or consumer code? Select 4 options.

- [ ] A. **Deliver real-time streaming data to Amazon Simple Storage Service (Amazon S3)**

- B. Deliver real-time streaming data to DynamoDB to support processing of digital documents
- C. Deliver real-time streaming data to Redshift to support data warehousing and real-time analytics
- D. Ingest data into ES domains to support Enterprise search built on Elasticsearch
- E. Allow Splunk to read and process data stream directly from Kinesis Firehose
- F. Ingest data into Amazon EMR to support big data analytics

Explanation:

Answer: A, C, D, E

Amazon Kinesis Data Firehose is a fully managed service for delivering real-time streaming data to destinations such as Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Elasticsearch Service (Amazon ES), and Splunk. With Kinesis Data Firehose, you don't need to write applications or manage resources. Configure data producers to send data to Kinesis Data Firehose, and it automatically delivers the data to the destination that you specified. You can also configure Kinesis Data Firehose to transform your data before delivering it.

- https://docs.aws.amazon.com/firehose/latest/dev/what-is-this-service.html

Question 7 **AWS Certified Big Data - Specialty**

Domain :Storage

Hymutabs Ltd (Hymutabs) is a global environmental solutions company running its operations in in Asia Pacific, the Middle East, Africa and the Americas. It maintains more than 10 exploration labs around the world, including a knowledge centre, an "innovative process development centre" in Singapore, a materials and membrane products development centre as well as advanced machining, prototyping and industrial design functions.
Hymutabs hosts their existing enterprise infrastructure on AWS and runs multiple applications to address the product life cycle management. The datasets are available in Aurora, RDS and S3 in file format. Hymutabs Management team is interested in building analytics around product life cycle and advanced machining, prototyping and other functions.
The IT team proposed Redshift to fulfill the EDW and analytics requirements. They adapt modeling approaches laid by Bill Inmon and Kimball to efficiently design the solution. The team understands that the data loaded into Redshift would be in terabytes and identified multiple massive dimensions, facts, summaries of millions of records and are working on establishing the best practices to address the design concerns.
There are 6 tables that they are currently working on:

- ORDER_FCT is a Fact Table with billions of rows related to orders
- SALES_FCT is a Fact Table with billions of rows related to sales transactions. This table is specifically used to generate reports EOD (End of Day), EOW(End of Week), and EOM (End of Month) and also sales queries
- CUST_DIM is a Dimension table with billions of rows related to customers. It is a TYPE 2 Dimension table
- PART_DIM is a part dimension table with billions of records that defines the materials that were ordered
- DATE_DIM is a dimension table
- SUPPLIER_DIM holds the information about suppliers the Hymutabs work with

SALES_FCT and DATE_DIM are joined together frequently since EOD sales reports are generated every day. please suggest your distribution style for both tables. Select 1 option.

A. **Distribute the SALES_FCT with KEY DISTRIBUTION on its own Primary KEY (one of the columns) while DATE_DIM is distributed with KEY DISTRIBUTION on Its PRIMARY KEY**

B. **Distribute the SALES_FCT with EVEN DISTRIBUTION on its own Primary KEY (one of the columns) while DATE_DIM is distributed with EVEN distribution on Its PRIMARY KEY**

C. **Distribute the SALES_FCT with KEY DISTRIBUTION on its own Primary KEY (one of the columns) while DATE_DIM is distributed with ALL DISTRIBUTION on Its PRIMARY KEY**

D. **Distribute the SALES_FCT with ALL DISTRIBUTION on its own Primary KEY (one of the columns) while DATE_DIM is distributed with EVEN distribution on Its PRIMARY KEY**

E. **Distribute the SALES_FCT with EVEN DISTRIBUTION on its own Primary KEY (one of the columns) while DATE_DIM is distributed with ALL distribution on Its PRIMARY KEY**

Explanation:

Answer : C

- **Option A is incorrect -** KEY DISTRIBUTION distributes the rows are according to the values in one column. This is a right approach to design the table, but DATE_DIM with KEY DISTRIBUTION with number of records being very low, lot of data is copied between nodes. This approach is ok but not a perfect design to build the solution

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option B is incorrect -** EVEN DISTRIBUTION evenly distributes the rows across the slices in a round-robin fashion, regardless of the values in any particular column. EVEN distribution is appropriate when a table does not participate in joins. For a fact table like SALES_FCT, all the nodes participate in all queries even though the EOD reports is only for that particular day

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option C is correct -** ALL distribution makes a copy of the entire table in every compute node. Being billion record tables, this is not a right approach to design. This is the perfect design for DATE_DIM table which has very low number and can be distributed to all tables

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option D is incorrect -** ALL distribution makes a copy of the entire table in every compute node. Being billion record tables, this is not a right approach to design. Cannot be used for massive table like SALES_FCT.

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html

- **Option E is incorrect -** EVEN DISTRIBUTION evenly distributes the rows across the slices in a round-robin fashion, regardless of the values in any particular column. EVEN distribution is appropriate when a table does not participate in joins. For a fact table like SALES_FCT, all the nodes participate in all queries even though the EOD reports is only for that particular day. SALES_FCT TABLE need to be designed on a table with a perfect distribution key in mind

https://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-distribution.html