1. Question on Memory Optimization

- **Category:** AIP – Operational Efficiency and Optimization for Generative AI Applications.
- **Scenario:** A generative AI chatbot on Amazon Bedrock experiences performance degradation and crashes due to **memory-related errors** when processing complex or high-volume queries simultaneously.
- **Question:** Which solution will resolve this issue and improve the model's stability by managing higher memory demands during peak usage?.
- **Options:**
    1. Add more processing power to the Bedrock AgentCore agent.
    2. Increase the number of Bedrock AgentCore agent instances.
    3. Upgrade the storage capacity of the Bedrock AgentCore agent.
    4. **Expand the memory capacity of the Bedrock AgentCore agent. (Correct Solution)**
- **Explanation: Agent Core memory** in Amazon Bedrock refers to the system's RAM used to process active data like input queries, model parameters, and intermediate results. When memory is insufficient, performance degradation and crashes occur. Expanding the memory capacity allows the system to handle larger datasets, more complex queries, and higher volumes without degradation, ensuring better stability and responsiveness. Increasing the number of instances is only a scaling solution and does not directly solve the insufficient memory issue.

---

2. Question on A/B Testing Deployment

- **Category:** AIP – Implementation and Integration.
- **Scenario:** A financial analytics firm needs to evaluate a **new fraud detection model's prediction accuracy and latency in production** without affecting the throughput of the currently deployed model or requiring any changes to how clients invoke the inference endpoint.
- **Question:** Which option satisfies the given requirements?.
- **Options:**
    1. Deploy both models in separate SageMaker endpoints and use Amazon CloudWatch metrics to compare their results in post-processing.
    2. Register the new model version in SageMaker Model Registry and configure an event trigger in AWS Lambda to automatically swap the endpoint to the new model after initial validation.
    3. Configure an Amazon API Gateway endpoint that splits traffic between the current and the new SageMaker endpoints for A/B testing.
    4. **Modify the existing SageMaker AI endpoint configuration by adding the new model as a ProductionVariant through the ProductionVariant API, and set a small InitialVariantWeight compared to the existing model's ProductionVariant VariantWeight to control the percentage of traffic routed to it. (Correct Solution)**
- **Explanation:** SageMaker's CreateEndpoint API allows defining multiple models as ProductionVariant instances for hosting. By setting an InitialVariantWeight, traffic can be distributed between the existing and new models (e.g., two-thirds to Model A, one-third to Model B). This enables real-time A/B comparison in production, minimizes operational overhead, and ensures the client invocation method remains unchanged.

---

3. Question on Low-Overhead Model Migration

- **Category:** AIP – Implementation and Integration.
- **Scenario:** A financial organization needs to migrate a custom, real-time fraud detection model (less than 5 GB, up to 50 concurrent requests) from on-premises infrastructure to AWS, prioritizing **minimal infrastructure management**.

- **Question:** Which solution meets the requirements with the least operational overhead?.
- **Options:**
    1. Deploy the custom fraud detection model in Amazon SageMaker Neo to optimize the model, then host the optimized model on a SageMaker real-time endpoint.
    2. Create a model configuration within Amazon SageMaker AI, then deploy the custom fraud detection model on an asynchronous SageMaker endpoint.
    3. Deploy the fraud detection model on a highly available Amazon EC2 instance in an auto-scaling group. Configure an application load balancer to route the incoming requests to the EC2 instance.
    4. **Create a model configuration within Amazon SageMaker AI, then deploy the custom fraud detection model on a serverless SageMaker endpoint. (Correct Solution)**
- **Explanation: SageMaker Serverless Inference** provides high availability and automatic scaling, operating on a pay-per-use model that scales down to zero when idle. It supports models up to 6 GB and a maximum concurrency of up to 200 per endpoint, perfectly suiting the real-time, low-volume fraud detection model with minimal infrastructure management required. Deploying to a real-time endpoint requires managing compute instances, increasing overhead. Asynchronous endpoints are for batch or long-running inference, not real-time use cases.

---

4. Question on Notebook Instance Security

- **Category:** AIP – AI Safety, Security, and Governance.
- **Scenario:** SageMaker notebook instances are deployed inside an isolated VPC with interface endpoints, yet unauthorized external users can still access them through the internet.
- **Question:** How can the team limit access to the SageMaker notebook instances, ensuring only authorized VPC users can connect?.
- **Options:**

1. Apply VPC Endpoint Policies to control which IAM users or services can access SageMaker AI through the VPC interface endpoint, providing more granular access control for interactions with SageMaker AI.
2. Set up VPC Traffic Mirroring to capture traffic to and from the notebook instances and identify unauthorized access attempts, enabling enhanced monitoring.
3. Update the security group for the notebook instances to restrict incoming traffic to only the CIDR blocks associated with the VPC. Apply this security group across all interfaces linked to the SageMaker notebook instances.
4. **Configure an IAM policy that allows sagemaker:CreatePresignedNotebookInstanceUrl and sagemaker:DescribeNotebookInstance actions exclusively from VPC interface endpoints. Ensure this policy is applied to the appropriate IAM users, groups, and roles. (Correct Solution)**

- **Explanation:** To secure the notebook instances and block external access, an **IAM policy** must be used. By creating a policy that restricts API actions like sagemaker:CreatePresignedNotebookInstanceUrl and sagemaker:DescribeNotebookInstance with a condition requiring the request to originate from the specified VPC interface endpoint, unauthorized access over the internet is prevented.

---

5. Question on Generative AI Model Customization

- **Category:** AIP – Operational Efficiency and Optimization for Generative AI Applications.
- **Scenario:** A publishing company uses a text-to-text foundation model (FM) on Amazon Bedrock for summarization. The model misinterprets casual language, local expressions, and abbreviations in customer feedback, leading to inaccurate summaries.
- **Question:** Which solution provides the most efficient and cost-effective approach to improve the model's understanding of customer feedback?.
- **Options:**

1. Launch a new large-scale training job in Amazon SageMaker AI using the model-parallelism library to build a domain-specific language model trained entirely on historical customer reviews.
2. Use Amazon SageMaker Data Wrangler to preprocess customer feedback data, remove slang and abbreviations, and standardize the language before sending it to the Bedrock model for summarization.
3. Implement Custom Entity Recognition (CER) to extract slang terms and abbreviations from customer feedback and use these extracted entities as metadata inputs to Bedrock during text generation.
4. **Customize the current foundation model by applying fine-tuning using labeled datasets of customer feedback that reflect informal wording, abbreviations, and expressions**. **(Correct Solution)**

- **Explanation: Fine-tuning** adjusts the parameters of an existing foundation model to improve its understanding of domain-specific context. By using labeled examples of unique phrasing, the model learns to accurately interpret these specialized text patterns. This is highly efficient and cost-effective compared to training a new large model from scratch, requiring less data and fewer resources.

---

6. Question on Computer Vision Modeling

- **Category:** AIP – Operational Efficiency and Optimization for Generative AI Applications.
- **Scenario:** A research organization needs a custom deep learning model (built via SageMaker AI) to **identify animal species** and **precisely locate each one** in images, generating **bounding boxes**.
- **Question:** Which modeling approach should be implemented in SageMaker AI to both identify animal species and locate each one precisely within the images?.
- **Options:**
    1. Use a SageMaker AI algorithm for semantic segmentation to label every pixel in the image.
    2. Train a SageMaker AI algorithm for image classification to categorize the dominant species in an image without detecting specific locations.

3. Deploy a custom pose estimation model in SageMaker AI to identify animal skeletal keypoints and postures rather than detecting full animal objects or their locations.
4. **Use a SageMaker AI object detection algorithm to accurately identify and localize multiple animal species by generating bounding boxes and associated class labels. (Correct Solution)**

- **Explanation: Object detection** algorithms are designed to both classify and localize multiple objects within an image, producing bounding boxes and associated class labels. This approach directly meets the requirements for identifying multiple species (classification) and determining their exact positions (localization via bounding boxes). Image classification only provides a single label for the entire image.

---

7. Question on Fairness and Class Imbalance

- **Category:** AIP – Implementation and Integration.
- **Scenario:** A data scientist needs to develop a fraud detection model on SageMaker with a severely imbalanced dataset (fraudulent transactions are rare). They must minimize operational overhead and ensure the model is fair and unbiased.
- **Question:** Which approach will fulfill the given requirements?.
- **Options:**

1. Use SageMaker Studio to preprocess the data and apply SMOTE, then use SageMaker Reinforcement Learning to build a fraud detection model and check for bias with SageMaker Clarify.
2. Use SageMaker Studio for data processing and model development, integrating the synthetic minority oversampling technique (SMOTE) into the workflow. Once the model is trained, use Amazon Augmented AI (Amazon A2I) for bias detection before deployment.
3. Use SageMaker Studio to preprocess the data and apply the synthetic minority oversampling technique (SMOTE) to balance the dataset. Build the model using SageMaker Pipelines and use SageMaker Clarify for bias detection before deployment.

4. **Use SageMaker Studio to preprocess and balance the data using the synthetic minority oversampling technique (SMOTE), then develop a fraud detection model with SageMaker JumpStart. Afterward, use SageMaker Clarify to check for bias and finalize the model for deployment. (Correct Solution)**

- **Explanation:** This approach is low-code and low-overhead. The **Synthetic Minority Oversampling Technique (SMOTE)** addresses class imbalance by generating synthetic examples for the rare class. **SageMaker JumpStart** reduces development complexity by offering ready-to-use solutions. **SageMaker Clarify** is purpose-built to assess model fairness and interpretability by calculating bias metrics.

---

8. Question on IoT Data Ingestion Pipeline

- **Category:** AIP – Foundation Model Integration, Data Management, and Compliance.
- **Scenario:** An AI developer needs a scalable, secure way to collect telemetry data (temperature, pressure) from devices in remote locations with unstable connectivity, store it in Amazon S3, and minimize infrastructure management.
- **Question:** Which solution meets the given requirements?.
- **Options:**

1. Stream the telemetry data over Message Queuing Telemetry Transport (MQTT) to AWS IoT Core, forward it to an Amazon Kinesis Data Stream, and then configure an AWS Lambda function to process the data and send it to S3.
2. Set up a serverless application with Amazon API Gateway to collect telemetry data from the devices, then use AWS Lambda to process and deliver the data to S3.
3. Use AWS IoT Greengrass on each device to preprocess telemetry data locally, then batch upload the data to S3 using AWS SDK calls from the edge.
4. **Route telemetry data over Message Queuing Telemetry Transport (MQTT) to AWS IoT Core, configure a rule in IoT Core to direct the data to an Amazon Data Firehose stream that delivers data to an S3. (Correct Solution)**

- **Explanation: AWS IoT Core** is designed to handle device connections using the **MQTT** protocol. **IoT Core rules** can efficiently direct the data stream to **Amazon Data Firehose**, which is a fully managed, serverless service optimized for direct delivery of real-time streaming data to destinations like Amazon S3, requiring no application writing or resource management.

---

9. Question on PII/PHI Redaction (Select TWO)

- **Category:** AIP – AI Safety, Security, and Governance.
- **Scenario:** After text extraction (Textract), the company must identify and remove **PII and PHI** from clinical notes before storage in S3 to meet HIPAA and data privacy standards, while preserving medical details.
- **Question:** Which services should be used to automatically detect and redact both PII and PHI entities from the extracted clinical text? (Select TWO.).
- **Options:**
    1. Use Amazon SageMaker Ground Truth to manually annotate medical entities and remove PII and PHI before training generative AI models.
    2. **Use Amazon Comprehend to detect and redact personally identifiable information (PII) from the text extracted. (Correct Solution)**
    3. **Use Amazon Comprehend Medical to identify and redact protected health information (PHI) and extract clinically relevant entities such as medical conditions, medications, and treatments. (Correct Solution)**
    4. Use Amazon Transcribe to convert the clinical text into audio transcripts that can later be reviewed for sensitive data.
    5. Use Amazon Polly to convert the extracted medical text to speech, ensuring sensitive information is obscured through audio synthesis.
- **Explanation: Amazon Comprehend** handles general NLP and includes built-in capabilities for identifying and redacting generic **PII**. **Amazon Comprehend Medical** analyzes unstructured clinical text specifically to identify and redact **PHI** while extracting medical entities (diagnoses, medications, etc.). This combination ensures both PII and PHI are removed automatically for compliance.

---

10. Question on Data Visualization for Normalization

- **Category:** AIP – Foundation Model Integration, Data Management, and Compliance.
- **Scenario:** An ML engineer uses Amazon SageMaker Data Wrangler to explore a numerical feature (image brightness) before applying normalization, as it affects model convergence.
- **Question:** Which action should the engineer take to best understand the range and distribution of the brightness feature values before transformation?.
- **Options:**
    1. The engineer should use Comprehend to perform sentiment analysis on the brightness values to determine if normalization is needed.
    2. The engineer should export the dataset to Amazon S3 and use AWS Glue DataBrew to create a box plot visualization of the brightness feature.
    3. The engineer should use SageMaker Clarify to detect data bias in the brightness feature before performing any normalization.
    4. **The engineer should use the SageMaker Data Wrangler histogram visualization to inspect the range of values for the brightness feature and identify any outliers. (Correct Solution)**
- **Explanation: SageMaker Data Wrangler** supports various visualizations. A **histogram** is most effective for exploring numerical features because it visually represents how data points are distributed across value ranges, making it easy to identify outliers, skew, or patterns critical for normalization decisions.

---

11. Question on Secure Data Redaction and Encryption

- **Category:** AIP – AI Safety, Security, and Governance.
- **Scenario:** A financial company needs to train a predictive model using customer data that includes sensitive information like credit card numbers. The data must be protected (encrypted) and credit card numbers must be redacted before training.
- **Question:** Which solution meets this requirement?.
- **Options:**

1. Use SageMaker AI Data Wrangler to process the customer data and apply encryption using a custom algorithm. Store the encrypted data in Amazon S3, then train the model using the encrypted dataset.
2. Use SageMaker AI Principal Component Analysis (PCA) algorithm to reduce the dimensionality of the customer data before training the model. Ensure that sensitive information, like credit card numbers, is removed during the PCA process and the data is stored in Amazon S3.
3. Use Comprehend to detect and redact PII from the customer data, then store the redacted data in Amazon S3 and train the model using SageMaker AI. Encrypt the customer data using a custom algorithm.
4. **Encrypt customer data with AWS KMS in Amazon S3 before importing it into SageMaker AI. Use AWS Glue to redact credit card numbers from the customer data before using it for model training. (Correct Solution)**

- **Explanation:** Using **AWS KMS** provides secure, native encryption at rest in Amazon S3, meeting compliance requirements. **AWS Glue** includes a specialized **Detect PII transform** designed to identify, mask, or remove PII entities like credit card numbers before the data is used for model training.

---

12. Question on Retrieval-Augmented Generation (RAG)

- **Category:** AIP – Foundation Model Integration, Data Management, and Compliance.
- **Scenario:** A Bedrock chatbot uses Amazon Titan Text but provides generic answers because it lacks access to proprietary order management and product documentation data (S3, internal DB). The team needs to enhance responses using this private data **without retraining the model**.
- **Question:** Which option satisfies this requirement?.
- **Options:**

1. Increase the temperature parameter for the Amazon Titan model to improve contextual understanding.
2. Use the Comprehend custom classification to provide the model with retrieval capabilities.
3. Fine-tune the Amazon Titan foundation model with the company's support data using Amazon SageMaker AI.