CERTLABZ CERTIFICATION BOARD

Certified MLOps Professional (CMOP)

Ten Sample Questions

These ten questions are written at the difficulty of the live exam. Six carry a figure, a data table, a configuration or a code listing; four are plain scenario questions, as on the live paper. Every one is a scenario a working MLOps engineer would meet, and none of them appears in the live question bank.

The live exam is multistage adaptive: how you score on one stage selects the difficulty of the next, so no two candidates sit the same paper. It is a performance-based exam: each form can carry up to 5 to 12 performance-based questions alongside the multiple-choice questions, which means you build, sequence or complete something rather than only choosing from a list. Answers and full explanations follow each question here; the live exam does not disclose item-level answers.

Question 1

Domain 1: MLOps Foundations and the ML Lifecycle

CONFIG

Emma Wilson lists the ML lifecycle stages, shown out of order.

```
A. Monitor performance and drift
B. Prepare and version the data
C. Deploy and serve the model
D. Train and evaluate the model
```

Listing 1.1 Stages to order.

What is the correct order of these stages?

(A) A, then C, then D, then B, monitoring the model before the data has been prepared or the model has even been trained yet.

(B) B, then D, then C, then A, preparing data, training and evaluating, deploying and serving, then monitoring in production.

(C) D, then B, then A, then C, training before the data is prepared and monitoring before the model is deployed to production.

(D) C, then D, then B, then A, deploying before training and preparing the data only after the model is already serving traffic.

Correct answer: B

The ML lifecycle runs from preparing and versioning data (B), to training and evaluation (D), to deployment and serving (C), to monitoring for drift in production (A). The other orders monitor or deploy before the model has been trained or the data prepared.

Question 2 — Domain 1: MLOps Foundations and the ML Lifecycle — DATA

James Carter cannot reproduce a model a colleague trained last quarter.

Was it captured?	Item
No	The exact dataset version
Partly	The training code
No	The library versions used
No	The hyperparameters

Table 2.1 What was and was not captured.

What MLOps practice would have made the model reproducible?

A. Buying a larger GPU, because reproducibility is a matter of compute power rather than of what was recorded about the run.

B. Versioning the data, code, environment and parameters together, so the exact run can be recreated later on demand.

C. Training the model for more epochs, because a more thoroughly trained model is inherently easier to reproduce afterwards.

D. Storing only the final model file, because keeping the trained weights is all that is ever needed to reproduce a result.

Correct answer: B
Reproducing an ML run requires versioning the data, code, environment and parameters together, none of which were fully captured here. A bigger GPU, more epochs, or keeping only the weights does not let you recreate how the model was produced.

Question 3 — Domain 2: Data and Feature Pipelines — CONFIG

Olivia Davis investigates a model that scores well offline but poorly live.

```
training: feature computed with a pandas script
serving : feature reimplemented in the API service
result  : feature values differ subtly between the two
```

Listing 3.1 How the feature is computed.

What is the problem, and what is the fix?

A. The model is too small, so the fix is to train a larger model that is robust to the small differences in the feature values.

B. Training-serving skew from computing the feature two different ways; share one feature definition across training and serving.

C. The offline metrics were simply too optimistic, so the fix is to lower the reported offline accuracy to match the live results.

(D) The serving hardware is too slow, so the fix is to add more servers so the feature is computed faster during live inference.

Correct answer: B
Computing a feature one way in training and reimplementing it in serving produces training-serving skew, which is a classic reason a model does well offline but poorly live. The fix is to share one feature definition (for example through a feature store or shared library). Model size, reported metrics and server speed do not address the mismatch.

Question 4 — Domain 4: Deployment and Serving — DATA

Liam Anderson matches rollout strategies to what they do.

Strategy	Behaviour
Canary	Small share of traffic to the new model first
Shadow	Runs on live traffic without serving its results
Blue-green	Switch between two environments, instant rollback
Deploy to all at once	No safeguard

Table 4.1 Rollout strategies.

A new model must go live with minimal risk of a costly incident. Which approach fits?

(A) Deploy to all traffic at once, because pushing the new model everywhere immediately is the fastest way to realise its benefits.

(B) A canary (or shadow first) rollout with monitoring and automatic rollback, so problems are caught before they affect all users.

(C) Skip deployment safeguards entirely, because a model that passed offline evaluation cannot cause an incident once it is in production.

(D) Train a second identical model as a backup, because having a duplicate model removes the need for any staged rollout or monitoring.

Correct answer: B
A canary rollout (optionally shadow first) with monitoring and automatic rollback limits the blast radius of a bad model and catches problems before they hit everyone. Deploying to all at once or skipping safeguards is exactly what causes costly incidents, and a duplicate model is not a rollout strategy.

Question 5 — Domain 5: Monitoring, Drift and Retraining — DATA

Ava Thompson reviews a production fraud model over several months.

Observation	Detail
Input distribution	Gradually shifting
Fraud patterns	Evolving as fraudsters adapt
Accuracy	Slowly declining
Labels	Arrive weeks later

Table 5.1 Production observations.

What is happening, and what is the right response?

A. Nothing is wrong, because a model that was accurate at launch will remain accurate indefinitely regardless of how the world changes.

B. Concept drift as fraud patterns evolve; monitor for drift and retrain on recent, validated data through proper evaluation gates.

C. The serving hardware is failing, so replacing the servers will restore the model's accuracy without any need to retrain the model.

D. The labels are simply wrong, so the fix is to stop collecting ground-truth labels and rely on the model's own predictions instead.

Correct answer: B
Evolving fraud patterns changing the input-to-target relationship is concept drift, which slowly degrades the model. The response is to monitor for drift and retrain on recent, validated data behind evaluation gates. Models do degrade over time, the hardware is not the cause, and abandoning ground-truth labels would remove the ability to measure accuracy at all.

Question 6

Domain 6: Governance, CI/CD and Reliability

CONFIG

An automated pipeline retrains and deploys a model every night.

```
nightly:
  retrain on latest data
  deploy new model to production
  # no evaluation gate, no comparison to current model
```

Listing 6.1 The nightly pipeline.

What is the risk, and what should be added?

A. There is no risk, because a freshly retrained model is always at least as good as the model it replaces in production each night.

B. A degraded or biased model can silently reach production; add evaluation gates and comparison to the current model before promotion.

C. The only risk is cost, so the fix is to retrain less often while still deploying every new model straight to production without checks.

D. The pipeline should retrain even more frequently, because more frequent retraining removes the need to evaluate any individual model.

Correct answer: B
Auto-deploying every nightly retrain with no evaluation gate or comparison lets a degraded or biased model reach production silently. The fix is to add evaluation gates and a champion-challenger comparison so a new model is promoted only if it genuinely passes and improves. Retraining is not automatically an improvement, and changing the frequency does not address the missing gate.

Question 7

Domain 2: Data and Feature Pipelines

A model looked excellent offline but failed in production; investigation shows a training feature included information only known after the prediction time. What is this, and why does it matter?

A. It is a minor issue, because using any available information in training always makes the model stronger and more accurate in production.

B. It is data leakage: the training data used information not available at prediction time, so the offline metrics were unrealistically good.

C. It is training-serving skew, which is unrelated to the timing of information and is caused only by different serving hardware being used.

D. It is normal behaviour, because offline and online performance are expected to differ greatly and the gap requires no investigation at all.

Correct answer: B
Using information in training that would not be available at prediction time is data (label) leakage, which inflates offline metrics and then fails live. Point-in-time correctness prevents it. It is not a strength, it is distinct from serving-hardware issues, and a large offline-to-online gap is a red flag, not something to ignore.

Question 8

Domain 3: Training, Experiment Tracking and Model Registry

A newly promoted model degrades in production and the team needs the previous model back within minutes. What makes this possible?

A. Retraining the previous model from scratch, because rebuilding it fresh is the fastest and most reliable way to restore production service.

B. A model registry with versioning and a tested rollback, so the prior known-good model version can be restored to production quickly.

C. Buying a larger GPU, because more compute is what allows a degraded production model to be swapped out for a better one in minutes.

D. Collecting more training data, because a bigger dataset is what enables a fast rollback to the previously deployed model version.

Correct answer: B
A model registry that versions models and supports a tested rollback lets the team restore the previous known-good version in minutes. Retraining from scratch is far too slow for an incident, and more GPU or more data does nothing to enable a fast rollback.

Question 9 Domain 5: Monitoring, Drift and Retraining

Monitoring shows the input data has drifted, but measured accuracy still looks fine because labels arrive weeks later. What is the prudent action?

A. Ignore the drift, because the accuracy figure is currently fine and input drift on its own never affects a model's real performance.

B. Investigate the drift and prepare to retrain, since accuracy may drop once the delayed labels arrive and confirm the degradation.

C. Immediately delete the drift monitor, because a signal that disagrees with the current accuracy figure is by definition a false alarm.

D. Switch off monitoring until the labels arrive, because there is nothing useful a team can do about input drift before it is confirmed.

Correct answer: B
Input drift is a leading indicator that performance may fall once delayed labels arrive, so the prudent action is to investigate and prepare to retrain rather than wait for accuracy to confirm the problem. Ignoring the drift, deleting the monitor, or switching off monitoring all discard an early-warning signal.

Question 10 Domain 6: Governance, CI/CD and Reliability

A regulator asks how a specific automated decision was made by a model several months ago. What must be in place to answer?

A. Only the current production model, because whatever model is live today can always explain a decision that was made months in the past.

B. Versioned data, model and code plus decision logs, so the exact model and inputs behind that past decision can be reconstructed.

C. A larger, more accurate model, because higher accuracy is what allows an organisation to explain historical automated decisions to regulators.

D. Nothing, because automated decisions made in the past can never be explained once the model has been retrained or updated since then.

Correct answer: B
Answering a regulator about a past decision requires versioned data, model and code plus logs of the decision, so the exact model and inputs can be reconstructed and explained. The current model does not represent what was live months ago, a bigger model does not add traceability, and with proper lineage and logging historical decisions can indeed be explained.

Published by the CertLabz Certification Board, a division of CertLabz LLC. Companion to objectives document CMOP-001. https://certlabz.com/certifications/certified-mlops-professional.html