Congratulations! You passed!

Grade received 100% **To pass** 80% or higher

Go to next item

1. What determines the maturity of the MLOps process? **1 / 1 point**

- (●) The level of automation of ML pipelines.
- ( ) The technical scenario for MLOps implementation.
- ( ) The standard pattern for MLOps building.
- ( ) The accuracy of the deployed model.

Correct

Great job! Fundamentally, the level of automation of the Data Collection, Modeling, Deployment, and Maintenance systems determines the maturity of the MLOps process.

2. True Or False: At the basic level of maturity, or level 0, tracking or logging are required to detect model performance degradation and other model behavioral drifts.

1 / 1 point

- ○ True
- ◉ False

Correct

Exactly! Level 0 is concerned only with deploying the trained model as a prediction service (for example, a microservice with a REST API). This manual, data-scientist-driven process doesn't track predictions and might be sufficient when models are rarely changed or retrained.

3. What steps do you need to introduce into the ML pipeline to move towards MLOps maturity level one? (Select all that apply)

1 / 1 point

- [x] Automated Data Validation

Correct
That's right! Automated data validation is necessary to decide whether your model should be retrained or the execution of the pipeline must stop. This decision is automatically made only if the data is deemed valid.

- [ ] Dependently Typed Programming
- [x] Automated Model Validation

Correct
Well done! Model Validation makes sure that the new model performs better than the current one before promoting it to production. Also, Model Validation ensures that model performance is consistent on various segments of the data.

- [ ] Automated Theorem Proving

4. In case of an interruption, what key component of the pipeline allows you to resume execution seamlessly? **1 / 1 point**

- ○ Models Registry
- ○ Trigger
- ◉ Metadata Store
- ○ Feature Store

Correct

Nailed it! The metadata store tracks each pipeline execution, so you can rely on pointers to the artifacts produced at each step, like the location of prepared data, validation anomalies, or computed statistics, to resume execution in case of an interruption.