✅ Dashboards as Code – Quiz

---

1. Addressing Concerns About Time Investment

Question:
A cloud data analyst is asked to explain to the board of directors why the organization should adopt a dashboards as code approach. The board of directors is concerned about how time-consuming it can be to build entire dashboards using code. To address these concerns, which of the following points should the cloud data analyst highlight?

- ❌ The time to build a dashboard is relative and will depend on the complexity of the dashboard.
  This is true but doesn't directly address the concern.
- ❌ Dashboards following the dashboards as code approach are optimized for public distribution.
  This is not a key benefit of the approach.
- ✅ **Dashboards following the dashboards as code approach are reusable.**
  Reusability saves time in the long run and supports scalability.
- ❌ The time to build a dashboard decreases as the team gains experience.
  While true, it's not the strongest argument to reassure stakeholders.

📝 **Feedback:**
Highlighting **reusability** shows how the initial time investment pays off by enabling faster development and consistency across future dashboards.

---

2. Making Changes to a Live Dashboard

Question:
A cloud data analyst is asked to revise a live dashboard that was built using the dashboards as code approach. How should they make the changes?

- ❌ Reuse the code and create a new dashboard using code as the approach.
  Not necessary to create a new dashboard—just update the existing one.
- ❌ Update the live dashboard in the visualization tool's user interface.
  This bypasses the code-based workflow and version control.
- ✅ **Update the underlying code in a developer environment.**
  This is the correct and safe way to make changes in the dashboards-as-code approach.
- ❌ Reuse the visualization and create a new one using the visualization tool's user interface.
  Again, this doesn't follow the dashboards-as-code methodology.

📝 **Feedback:**
Changes should be made in the **developer environment** to allow for **testing, review, and version control** before going live.

---

3. Comparing with Visualization Tools

Question:
A data team is evaluating the adoption of a dashboards as code approach. They compare using a dashboards as code approach with using a visualization tool's user interface. Which of the following is a disadvantage of using a visualization tool's user interface?

- ❌ Changes are difficult to make for non-technical users.
  Actually, UIs are designed to be user-friendly.
- ❌ Dashboards take a long time to develop.
  UI-based dashboards are typically faster to build.
- ✅ **Changes are made directly over the dashboard live version.**
  This is a major drawback—errors go live immediately and are hard to roll back.
- ❌ Dashboards need to be created using predefined templates.
  This may be true for some tools, but it's not the main disadvantage.

📝 **Feedback:**
With UI-based dashboards, **changes go live immediately**, which increases the risk of errors and makes it harder to maintain quality and consistency.