CoGrammar

PORTFOLIO SESSION

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(FBV: Mutual Respect.)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Open Classes. You can submit these questions here: **Open Class Questions**

Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**
- Report a **safeguarding** incident: **www.hyperiondev.com/safeguardreporting**
- We would love your **feedback** on lectures: **Feedback on Lectures**

Prestigious Co-Certification Opportunities

New Partnerships!

- **University of Manchester & Imperial College London** join our circle along with The University of Nottingham Online.

Exclusive Opportunity:

- Co-certification spots awarded on a first-come basis.
- Meet the criteria early to gain eligibility for the co-certification.

New Deadlines:

- **11 March 2024**: 112 GLH & BYB tasks completion.
- **18 March 2024**: Record interview invitation or self-employment.
- **15 July 2024**: Submit verified job offer or new contract.

Recap

Encapsulation

- Encapsulate attributes and behaviour into a single object. Control access to data of object by making attributes and behaviour public, private or protected.

Abstraction

- Think of your objects at a high level. Rather determine what the object should do instead of how the object should do it.

Inheritance

- Use inheritance for object that share attributes and behaviour. This allows us to extend and change behaviour of our objects.

Polymorphism

- Allows objects of different classes to be treated as objects of a common superclass.

Which of the following diagrams show the relationships between my objects?

A. Sequence Diagram
B. Context Diagram
C. Class Diagram

Which SOLID principle states that a subclass should be substitutable for its base class?

A. Single Responsibility
B. Open-Closed
C. Liskov
D. Interface Segregation
E. Dependency Inversion

Open Source Contributions with Git

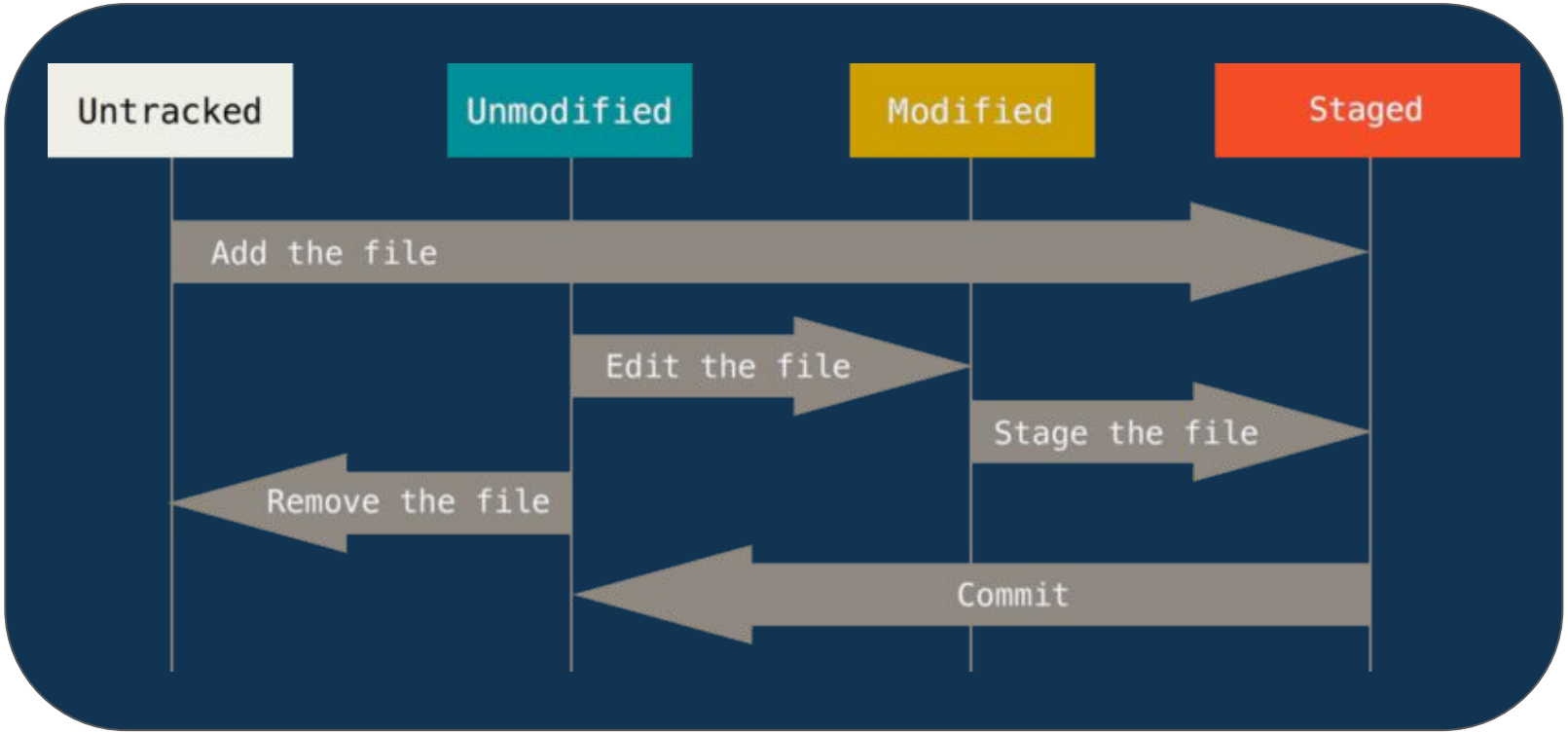
- **Background:** The open-source community is a testament to the collaborative spirit of software development, where ideas flourish and knowledge is freely shared. "CodeHaven" stands as a beacon for this ethos, inviting developers from all walks of life to contribute and learn.

- **Challenge:** Alex, a developer with a keen interest in mentoring, presents the you with a challenge that serves as a rite of passage into the open-source world. The task is to contribute to "FirstContributions," (*firstcontributions/first-contributions: 🚀✨ Help beginners to contribute to open source projects (github.com)*) a welcoming project for novices in the open-source community.

- **Objective:**
  Your task is to master the tools of the trade: Git and GitHub. You'll learn to navigate code repositories, manage updates, and understand the social coding environment that GitHub fosters. Through this hands-on experience, you won't only just contribute code, but also craft a narrative of your development journey.

  Your contributions will be your legacy, a digital portfolio that speaks to your ability to engage with complex workflows and collaborative projects. It's a chance to turn "CodeHaven" into a showcase of your dedication, skill, and readiness to join the global developer community!

Before Pushing



https://git-scm.com/book/en/v2/images/lifecycle.png

Demo: Cloning & Checking out

```
# Simple example to demonstrate the need for version control and
collaboration
# (Demonstration may include additional features not covered in the
example)

# Clone a repository from GitHub
git clone https://github.com/firstcontributions/first-contributions.git

# Create a new branch for a new feature or bug fix
git checkout -b my-feature
```

Demo: Staging, committing and Pushing

```
# Make changes to the code
# (Add, edit, or delete files)

# Stage changes for commit
git add .

# Commit changes with a meaningful message
git commit -m "Add my feature to the project"

# Push changes to the remote repository
git push origin my-feature
```

Demo: Markdown(.md)

Heading	# **H1** ## **H2** ### **H3**
Bold	**bold text**
Italic	*italicized text*
1. Ordered List	1. First item 2. Second item 3. Third item
• Unordered List	- First item - Second item - Third item

Demo: ReadMe

```
# Git Contribution Case Study

## Overview
Welcome to the Git Contribution Case Study!

## Getting Started

### Prerequisites

Before you begin, please ensure that you have the following installed on your machine:
- [Git](https://git-scm.com/)
- [GitHub Account](https://github.com/)

### Clone the Repository:

```bash
git clone https://github.com/firstcontributions/first-contributions.git
cd first-contributions
```
```

Git Contribution Case Study

Overview

Welcome to the Git Contribution Case Study!

Getting Started

Prerequisites

Before you begin, please ensure that you have the following installed on your machine:

- Git
- GitHub Account

Clone the Repository:

```
git clone https://github.com/firstcontributions/first-contributions.git
cd first-contributions
```

CodeHaven

Your task is to master the tools of the trade: Git and GitHub. You'll learn to navigate code repositories, manage updates, and understand the social coding environment that GitHub fosters.

Crucial git commands:

Git Status & Git Init
Git Fetch & Git Pull
GIt Add & Git Commit
Git Push

Important Concepts:

1. **README for project overview:** ReadMe markdown files are important, not only for providing context to your projects in terms of descriptions, but also for serving as a guide for people viewing your projects.
2. **Remember to Pull!:** It's important to always remember to pull any changes that have been made to your remote repository, before performing and pushes from you local repository.
3. **Documentation:** Remember to document your code using docstrings and comments.

Advanced Challenge:

- Start reviewing the code for any component of an open source project and consider contributing, by identifying bugs or potential user experience issues.

In Git, what does "git merge" command do?

A. Delete a branch
B. Create a new branch
C. Combine changes from one branch into another
D. Push changes to remote repository

Which command is used to view the commit history in Git?

A. git log
B. git status
C. git diff
D. git branch

Summary

Git and GitHub

- ★ Try to get comfortable with git repositories and using it alongside GitHub to work with remote repositories.

Contribute

- ★ After getting comfortable with the GitHub workflow try contributing to open source projects.

Questions and Answers

Questions around the Case Study

CoGrammar

Thank you for joining