



BridgeLabz

Employability Delivered

Objectives,
Expectations and
Rules for Logical
Programs and GIT
Usage

A large, abstract blue ink splash or watercolor blotch occupies the left side of the slide, extending from the top left towards the center. It has irregular, feathered edges and some darker blue areas within it.

Objectives

1. Ability to work with Bash Shell Scripts Programming Constructs
2. Ability to work with basic git flows.



GIT Flow Expectations

- Create Repo for Shell Scripts
- Add and Commit Files
- Follow Message Hygiene
- During Commit specify Add or Refactor in the message beginning
- Create Branch
- Push to Remote Master and Branch
- Merge and Resolve Conflicts
- Ability to look into History of Files
- Comfortable with Git Commands

Shell Programming Expectations

- Use of Shell Programming Constructs
- Use of proper names for the File Names, Variables and Constants
- Use of Proper Indentations
- Avoiding Printing to standard terminal instead use debug
- No Commented Codes
- Follow DRY Principle – Do not Repeat Yourself
- Comfortable with Debug Execution Threads
- Comfortable with Shell Commands



Rules

- Create empWageComputation.sh
- Start with Welcome message in the Main Branch
- Every Use Case (UC) in the Corresponding UC Branch
- For e.g. UC-1 Branch Name – EmployeeAttendanceUC
- Follow Programming Hygiene and DRY principle
- Testing the Program before pushing to Remote
- On Completion of every UC, do the following
 - Merge Local Branch with Local Master
 - Push to Remote Branch
 - Merge Local Master with Local Branch
 - Push Local Master to Remote Master



START

Start with Displaying
Welcome of the Program
being solved in the Master
Branch



UC TEMPLATE

Use Case in its own
branch and after
successful
completion merged
to Master

- Hints are given here



BridgeLabz

Employability Delivered

Thank
You