

- 1) **Quale delle seguenti opzioni descrive il Github Flow, il flusso di sviluppo "standard" su Github?**
  - a) Un metodo di sviluppo basato su branch, fork e merge
  - b) Un metodo di sviluppo basato su branch, push e merge
  - c) Un metodo di sviluppo basato su branch, pull request e merge
- 2) **Quale comando Git permette di unire due branch insieme?**
  - a) git combine
  - b) git join
  - c) git merge
- 3) **Come funziona il meccanismo di "branching" in Git?**
  - a) Il meccanismo di "branching" consente di condividere il codice con altri sviluppatori
  - b) Il meccanismo di "branching" consente di lavorare su più versioni del codice contemporaneamente
  - c) Il meccanismo di "branching" consente di creare una copia del codice per scopi di backup
- 4) **Qual è il ruolo principale di un Project Manager in un team di sviluppo software?**
  - a) Condurre analisi di mercato e definire la roadmap del prodotto
  - b) Scrivere codice e risolvere problemi tecnici
  - c) Coordinare le attività del team
- 5) **Qual è la principale differenza tra Git e GitHub?**
  - a) Non c'è differenza tra Git e GitHub
  - b) Git è un sistema di controllo versione, mentre GitHub è una piattaforma di hosting per progetti Git
  - c) Git è una piattaforma di hosting per progetti, mentre GitHub è un sistema di controllo versione
- 6) **Qual è lo scopo di un sistema di controllo versione come Git?**
  - a) Entrambe le altre risposte sono corrette
  - b) Tracciare i cambiamenti nel codice sorgente
  - c) Condividere il codice con altri sviluppatori

- 7) Quale delle seguenti opzioni è un vantaggio dell'utilizzo di una infrastruttura cloud?**
- a) Accentramento
  - b) Scalabilità
  - c) Entrambe le altre risposte sono corrette
- 8) Qual è il passo successivo dopo aver scritto e testato il codice in un ciclo di sviluppo software?**
- a) Deployment
  - b) Analisi
  - c) Testing
- 9) Qual è la principale differenza tra un branch "Main" e un "feature branch" in Git?**
- a) Il branch Main è utilizzato per nuove funzionalità, mentre un feature branch è utilizzato per il codice stabile
  - b) Non c'è differenza tra un branch "Main" e un "feature branch" in Git
  - c) Il branch Main è utilizzato per il codice stabile, mentre un feature branch è utilizzato per nuove funzionalità
- 10) Quale comando Git permette di creare un nuovo branch da un repository esistente?**
- a) git checkout
  - b) git branch
  - c) git clone
- 11) Quale comando Git permette di inviare i cambiamenti al repository remoto?**
- a) git push
  - b) git pull
  - c) git commit
- 12) Qual è il ruolo principale di un tester in un team di sviluppo software?**
- a) Scrivere codice sia lato server che lato client
  - b) Coordinare le attività del team e gestire i budget
  - c) Verificare la qualità del codice e segnalare eventuali problemi

**13) Cos'è un Dockerfile?**

- a) Un file di configurazione per i volumi di un container Docker
- b) Un file di log per un container Docker
- c) Un file di testo che contiene le istruzioni per creare un'immagine Docker

**14) Qual è il processo per creare una pull request su GitHub?**

- a) Creare un nuovo branch, fare delle modifiche al codice, fare una pull request per unire il nuovo branch al branch principale.
- b) Fare una pull request per unire il nuovo branch al branch principale, creare un nuovo branch, fare delle modifiche al codice.
- c) Fare delle modifiche al codice, creare un nuovo branch, fare una pull request per unire il nuovo branch al branch principale.

**15) Qual è la principale differenza tra un framework e una libreria?**

- a) Un framework è solo per il frontend, mentre una libreria è solo per il backend.
- b) Un framework è solo per il backend, mentre una libreria è solo per il frontend.
- c) Un framework fornisce una struttura predefinita per lo sviluppo del software, mentre una libreria fornisce solo un insieme di funzioni e classi che possono essere utilizzate per lo sviluppo del software.

**16) Qual è la principale differenza tra backend e frontend?**

- a) Il backend si occupa dell'interfaccia utente e dell'interazione con l'utente, mentre il frontend si occupa delle funzionalità del software che sono eseguite sul server.
- b) Il backend si occupa delle funzionalità del software che sono eseguite sul server, mentre il frontend si occupa dell'interfaccia utente e dell'interazione con l'utente.
- c) Il backend e il frontend sono la stessa cosa, si occupano dello sviluppo del software in generale

**17) Quando si inizia un nuovo progetto software, quali sono le principali decisioni da prendere?**

- a) Scelta del sistema operativo, scelta delle tecnologie di virtualizzazione, configurazione delle reti e sicurezza.
- b) Scelta del linguaggio di programmazione e di eventuali framework, scelta del database, configurazione dei server e scelta delle piattaforme cloud.
- c) Scelta del team di sviluppo, definizione dei ruoli, definizione della roadmap del prodotto e pianificazione delle scadenze.
- d) Tutte le altre risposte sono corrette

**18) Come il Product Owner lavora con il team di sviluppo?**

- a) Il Product Owner lavora con il team di sviluppo per scrivere codice e risolvere problemi tecnici.
- b) Il Product Owner lavora con il team di sviluppo per definire i requisiti del prodotto, prioritizzare i task e fornire supporto per risolvere problemi.
- c) Il Product Owner lavora con il team di sviluppo per gestire i budget e coordinare le attività del team.

**19) Quale è la principale differenza tra git clone e git pull?**

- a) git clone e git pull sono la stessa cosa
- b) git clone crea una copia del repository remoto sul proprio computer, mentre git pull crea una copia del repository remoto in una nuova cartella.
- c) git clone crea una copia del repository remoto sul proprio computer, mentre git pull aggiorna una copia esistente del repository

**20) Qual è il primo passo per iniziare un nuovo progetto utilizzando il Github Flow?**

- a) Creare un nuovo branch dal Main
- b) Creare un nuovo repository
- c) Creare una fork del repository originale

**21) Qual è il passo successivo dopo aver creato un nuovo branch in Github Flow?**

- a) Fare delle modifiche al codice nel nuovo branch
- b) Fare una pull request per unire il nuovo branch al Main
- c) Fare un merge del nuovo branch con il main

**22) Cos'è Docker?**

- a) Un sistema di gestione dei pacchetti per Linux
- b) Una piattaforma di virtualizzazione che consente di eseguire container
- c) Un sistema di gestione dei database

**23) Cos'è un container Docker?**

- a) Un'immagine Docker non ancora in esecuzione
- b) Un'istanza di un'immagine Docker in esecuzione
- c) Un file di configurazione per un'immagine Docker

**24) Come funziona l'architettura di Git per gestire le versioni del codice?**

- a) Utilizzando un sistema di gestione dei file centralizzato
- b) Utilizzando un sistema di gestione dei file distribuito
- c) Utilizzando un sistema di gestione dei file basato su cloud

**25) Come funziona il meccanismo di "commit" in Git?**

- a) Il meccanismo di "commit" registra una copia del codice in un punto specifico nel tempo
- b) Il meccanismo di "commit" invia il codice a un server remoto
- c) Il meccanismo di "commit" rende il codice disponibile per il deployment