

## 2.2 GIT зсередини

Git зберігає дані в репозиторії, який складається з комітів, гілок та тегів. Git використовує два типи команд:

- внутрішні системні команди, такі як `hash-object`, `cat-file` та інші, та
- команди користувача, що являють собою інтерфейс користувача.

### Об'єкти Git

Git зберігає дані в репозиторії у вигляді об'єктів.

Існує три типи об'єктів:

- Це файл або **blob** (великий бінарний об'єкт)
- **tree**: це об'єкт, який містить список інших об'єктів `tree` та `blob`, що представляють дерево файлів та папок.
- **commit**: це об'єкт, який містить інформацію про коміт, включаючи посилання на дерево файлів, автора коміту та повідомлення про коміт.

### Хеш-функції Git

Git використовує хеш-функції для ідентифікації об'єктів у репозиторії. Хеш-функція приймає на вхід дані та повертає унікальний ідентифікатор для цих даних. У Git використовуються SHA-1 та SHA-256 хеш-функції.

### Команда `hash-object`

Команда `hash-object` дозволяє створювати новий об'єкт Git та повертати його хеш-ідентифікатор. Команда може бути використана для створення нового `blob` об'єкту з вмістом файла:

```
$ echo "Hello, World!" | git hash-object -w --stdin
```

Ця команда створює новий `blob` об'єкт з вмістом "Hello, World!" та повертає його хеш-ідентифікатор.

### **Команда cat-file**

Команда cat-file дозволяє переглядати вміст об'єктів Git. Вона приймає тип об'єкту та його хеш-ідентифікатор та повертає вміст об'єкту. Наприклад, команда cat-file може бути використана для перегляду вмісту blob об'єкту:

```
$ git cat-file -p 83baae61804e65cc73a7201a7252750c76066a30
```

Ця команда повертає вміст blob об'єкту з хеш-ідентифікатором 83baae61804e65cc73a7201a7252750c76066a30.

### **Команда update-index**

Команда update-index дозволяє додавати або видаляти файли з індексу Git. Індекс Git - це структура даних, що містить інформацію про файли в робочій директорії та їх стан.

**Додавання нового файлу до індексу виконується за допомогою команди:**

```
$ git update-index --add path/to/file
```

Ця команда додає файл з шляхом path/to/file до індексу Git.

**Видалення файлу з індексу виконується за допомогою команди:**

```
$ git update-index --remove path/to/file
```

Ця команда видаляє файл з індексу Git.

### **Команда ls-files**

Команда ls-files дозволяє переглядати список файлів, які знаходяться в індексі Git. Вона може бути використана для перевірки, які файли були додані до індексу. Наприклад, команда ls-files може бути використана для перегляду списку файлів в індексі:

```
$ git ls-files
```

Ця команда виводить список файлів, які знаходяться в індексі Git.

### **Команда write-tree**

Команда write-tree дозволяє створювати новий tree об'єкт зі списком файлів та папок, які знаходяться в індексі Git. Команда може бути використана для створення нового tree об'єкту з поточного стану індексу:

```
$ git write-tree
```

Ця команда створює новий tree об'єкт зі списком файлів та папок, які знаходяться в індексі Git, та повертає його хеш-ідентифікатор.

### **Команда commit-tree**

Команда commit-tree дозволяє створювати новий commit об'єкт та пов'язувати його зі змінами, які були зроблені в індексі Git. Команда може бути використана для створення нового commit об'єкту з поточного стану індексу:

```
$ git commit-tree <tree-hash> -p <parent-commit-hash> -m "Commit message"
```

Ця команда створює новий commit об'єкт, який має дерево файлів з хеш-ідентифікатором <tree-hash>, батьківський коміт з хеш-ідентифікатором <parent-commit-hash> та повідомленням про коміт "Commit message".

Рекомендується для ознайомлення [Git зсередини](#)