Answer the following:

1. **Explain the concept of Projection.**

Ans –

Projection

Projection in computer science and software development refers to the process of selecting specific attributes or fields from a dataset or object. It is often used to retrieve only the required data rather than fetching entire records, optimizing performance and memory usage.

Types of Projections:

- **Interface-Based Projection:**
  In Spring Data, you can define an interface that specifies the fields you want to retrieve from the database. Spring automatically maps the results to the interface.

  Example:

```
public interface UserProjection {
   String getName();
   String getEmail();
}
```

- **DTO-Based Projection:**
  Custom classes (DTOs) are used to hold the projected data. These classes are mapped to the database query results, often using a constructor.

  Example:

```
public class UserDTO {
   private String name;
   private String email;

   public UserDTO(String name, String email) {
      this.name = name;
      this.email = email;
```

```
            }
        }
```

2. **Explain the concept of the PagingAndSortingRepository interface.**

Ans –

The PagingAndSortingRepository interface in Spring Data is a repository abstraction that provides methods for pagination and sorting of data. It extends the CrudRepository interface, which offers basic CRUD operations.

Key Features:

- **Pagination Support:**
  It allows fetching data in chunks (pages) rather than loading all records at once.
  Example:

  ```
  Page<User> findAll(Pageable pageable);
  ```

- **Sorting Support:**
  It supports sorting the data based on one or more properties.
  Example:

  ```
  Iterable<User> findAll(Sort sort);
  ```

Example :

```
@Autowired
private PagingAndSortingRepository<User, Long> userRepository;


public void paginateAndSortUsers() {
   Pageable pageable = PageRequest.of(0, 5, Sort.by("name").ascending());
   Page<User> users = userRepository.findAll(pageable);


   users.forEach(System.out::println);
}
```