

Progetto Spring Impiegati



JPA, 1 sola entity

Descrizione

- ▶ Si vuole realizzare un progetto Spring Boot per la gestione degli impiegati di una azienda.
- ▶ L'impiegato è definito da
 - ▶ nome
 - ▶ cognome
 - ▶ matricola
 - ▶ salario mensile
 - ▶ data assunzione

Funzionalità per gli impiegati

- ▶ Assumi nuovo impiegato
- ▶ Cerca impiegato per matricola
- ▶ Visualizza tutti gli impiegati
- ▶ Cancella un impiegato tramite la matricola e ritornalo
- ▶ Modifica il salario di un impiegato (passando la matricola e il nuovo salario) e ritorna l'impiegato aggiornato

AVANZATE:

- ▶ Cancella un impiegato tramite la matricola e torna solo nome e cognome
- ▶ Visualizza la lista dei nominativi (nome e cognome) degli impiegati ordinati per matricola
- ▶ Visualizza il totale dei salari che l'azienda paga ogni mese.
- ▶ Visualizza tutti gli impiegati ordinati per cognome
- ▶ Visualizza l'impiegato assunto da maggior tempo.
- ▶ Visualizza l'impiegato con salario maggiore assunto dopo una data specificata

Architettura

- ▶ *Gestire la persistenza con MySQL server e JPA/Hibernate*
- ▶ Utilizzare Tomcat Application Server per i servizi **REST**
- ▶ Strutturare il progetto con architettura **MVC**, prevedendo i seguenti package:
 - ▶ controller
 - ▶ service
 - ▶ dao
 - ▶ entity
 - ▶ dto

Configurazioni

- ▶ Nome pacchetto principale: **com.spring.impiegati**
- ▶ Definizioni nell' application.properties:

```
server.port=8080
```

```
spring.mvc.servlet.path=/gestioneimpiegati
```

```
spring.datasource.url=jdbc:mysql://localhost:3306/xxxx?createDatabaseIfNotExist=true&autoReconnect=true&useSSL=false
```

```
spring.datasource.username=xxxx
```

```
spring.datasource.password=xxxx
```

```
spring.jpa.hibernate.ddl-auto=update
```

```
spring.jpa.show-sql=true
```

Personalizzare le credenziali di MySQL