

DOCUMENTATIE

Introducere

Aplicatie prevede impartirea a 3 dealership-uri de masini in 3 baze de date, acestea sunt numite dupa numele firmei lor de dealershipuri (Dacia, Man etc). Apoi programul prevede manipularea datelor din cele 3 baze de date create.

Cerinta:

#Vehicule

3 baze de date

1 entitate: vehicul(id, nume, cai putere, an_fabricatie, motor, pret)

Programul administreaza dealership-uri de vehicule impartite in 3 categorii (una per baza de date, ex: dacia, man, volkswagen)

Functionalitate: se va permite afisarea vehiculelor(impreuna cu datele acestora), sortate crescator si descrescator, in functie de pret.

Functionalitate: se va permite adaugarea de vehicule noi (selectare manuala a datelor vehiculului)

Functionalitate: se va permite stergerea de vehicule

Exemplu de verificare: porneste programul, afiseaza toate vehiculele, verifica in PGAdmin, adauga cateva vehicule noi, verifica in PGAdmin, afiseaza vehiculele dupa pret, crescator sau descrescator

Conea Emanuel-Cosmin 10F271

Instalare PostgreSQL

1. Se va descarca PostgreSQL de la urmatorul link: <https://www.postgresql.org/download/> .
2. Se urmaresc pasii din wizard si se instaleaza componentele necesare:
 - PostgreSQL Server
 - pgAdmin 4
 - Stack Builder
 - Command Line tools
3. Se continua instalarea.

Instalare Java

1. Se va descarca Java de la urmatorul link: <https://www.java.com/en/download/win10.jsp>.
2. Se ruleaza installer-ul descarcat si se urmeaza pasii din wizard.
3. "Edit the system environment variables" -> System Properties (Tabul Advanced) se da click pe Environment Variables.
4. Se adauga o noua variabila de sistem accesand butonul New cu numele 'JAVA_HOME' si cu valoarea pathului unde am descarcat java.
5. In Variabilele de sistem se cauta 'Path' si se acceseaza butonul 'Edit' -> 'New' si se adauga path-ul catre folderul bin: %JAVA_HOME%\bin

Instalare Eclipse IDE for Java EE Developers

1. Se va descarca installer-ul de la urmatorul link:
<https://www.eclipse.org/downloads/download.php?file=/oomph/epp/2019-03/R/eclipse-instwin64.exe>
2. Se ruleaza instalerul descarcat in pasul anterior.
3. Vom da click pe optiunea Eclipse IDE for Enterprise Java Developers.
4. Se apasa butonul de install.
5. Se finalizeaza setup-ul.

Instalare Maven

1. Se acceseaza site-ul web <https://maven.apache.org/download.cgi> si se descarca arhiva apache-maven-3.6.1-bin.zip si se dezarhiveaza intr-o locatie aleasa.
2. Se da click pe iconita de search si se intra in "edit the system environment variables"
3. Dupa ce se deschide System Properties (Tabul Advanced) se da click pe Environment Variables.
4. Se adauga o noua variabila de sistem accesand butonul New cu numele 'MAVEN_HOME' si cu valoarea pathului unde au fost dezarhivate fisierele.
5. In Variabilele de sistem se cauta 'Path' si se acceseaza butonul 'Edit' -> 'New' si se adauga path-ul catre folderul bin: %MAVEN_HOME%\bin
6. Pentru verificare se ruleaza Command Prompt si se executa comanda : mvn -v

Outupul ar trebui sa fie asemanator cu acesta:

Apache Maven 3.6.1 (d66c9c0b3152b2e69ee9bac180bb8fcc8e6af555; 2019-04-04T22:00:29+03:00)

Maven home: D:\Programs\apache-maven-3.6.1\bin\..

Java version: 1.8.0_211, vendor: Oracle Corporation, runtime:
D:\App\Progra~1\Java\jdk1.8.0_211\jre

Default locale: en_US, platform encoding: Cp1250

OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

Despre proiect

1. Se deschide Command Prompt.
2. Se creeaza proiectul maven folosind comanda : mvn archetype:generate -DgroupId=com.java DartifactId=VehiclesJPA-DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false.
3. Se converteste proiectul Maven la unul Eclipse folosind comanda : mvn eclipse:eclipse cu Command Prompt deschis la locatia unde a fost creat proiectul.
4. In Eclipse IDE se alege File -> Import -> General -> Existing Projects into Workspace -> Choose your project folder location -> Done .
5. Se creeaza un folder numit 'resources' sub directorul 'src/main'.
6. Se va adauga directorul 'resources' in Build Path: Click-Dreapta pe proiect -> 'Build Path' -> 'Configure Build Path' -> Click pe tabul 'Source' -> Click pe 'Add Folder'.
7. Pe ecranul urimator se selecteaza casuta din stanga 'resources' si se apasa 'OK'.

8. Se adauga JPA si hibernate ca dependente in pom.xml.

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4 0 0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.java</groupId>
  <artifactId>VehiclesJPA</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>VehiclesJPA</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.eclipse.persistence</groupId>
      <artifactId>javax.persistence</artifactId>
      <version>2.0.0</version>
    </dependency>

    <dependency>
      <groupId>org.hibernate</groupId>
```

```

    <artifactId>hibernate-entitymanager</artifactId>

    <version>4.2.8.Final</version>

</dependency>

<dependency>

    <groupId>org.postgresql</groupId>

    <artifactId>postgresql</artifactId>

    <version>42.2.5</version>

</dependency>

</dependencies>

</project>

```

9. Se executa comanda mvn eclipse:eclipse in acelasi cmd deschis anterior.
10. Se creeaza folderul 'META-INF' in directorul numit 'resources' (adaugat anterior).
11. Se va crea 'persistence.xml' in interiorul folder-ului creat la pasul anterior.
12. Se va configura persistence.xml in functie de numarul bazelor de date:
13. Se vor crea urmatoarele clase:
 - Dao – Interfata generica prin care se fac operatiile CRUD pe o baza de date
 - DBHelper– clasa in care este definita conexiunile la bazele de date
 - Vehicle– clasa generata automat : Click-Dreapta pe proiect -> JPA Tools -> Generate Entities from Tables
 - VehicleDao– clasa in care sunt suprascrise metodele din Dao
 - VehicleController – clasa in care avem ca instanta un dao si metode
 - MainMenu– in aceasta clasa, avem mai multe optiuni, in functie de ceea ce alegem, se executa anumite operatii
 - Main – EntryPoint pentru program
14. Se va adauga JPA in proiect: Click-Dreapta proiect -> Properties -> Convert to Facet Form...>Select JPA -> Apply and Close

Adnotari JPA

Categorie Descriere Adnotare

Categorie	Descriere	Adnotare
Entitate	TopLink JPA presupune ca o clasa Java nu este persistenta si nu este eligibila pentru serviciile JPA decat daca este decorata cu aceasta adnotare utilizata pentru a desemna o clasa obiect obisnuita ca entitate, astfel incat sa poata fi utilizata cu serviciile JPA.	@Entity
Identitate	Fiecare entitate trebuie sa aiba cel putin un camp/o proprietate care serveste drept cheie primar. Concomitent, aceasta adnotare se poate utiliza pentru a ajusta modul in care baza de date pastreaza identitatea entitatilor.	@Id
Interogari	Intr-o aplicatie, se poate utiliza un manager de entitati pentru a crea si executa interogari in mod dinamic sau se pot predefini interogari si se pot executa dupa nume.	@NamedQuery