

Accolite University Feb Batch 1

Hibernate 2 Assignment

Ecommerce Associate Mapping

- Meet Shah
(INT626)

SCREENSHOTS

1. All Queries

```
<terminated> EcommerceManager [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (27-Feb-2021, 11:32:52 pm)
Feb 27, 2021 11:32:53 PM org.hibernate.Version logVersion
INFO: HHH0000412: Hibernate ORM core version 5.4.28.Final
Feb 27, 2021 11:32:53 PM org.hibernate.annotations.common.reflection.java.JavaReflectionManager <clinit>
INFO: HCAN0000001: Hibernate Commons Annotations (5.1.2.Final)
Feb 27, 2021 11:32:54 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl configure
WARN: HHH0001002: Using Hibernate built-in connection pool (not for production use!)
Feb 27, 2021 11:32:54 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH0001005: using driver [com.mysql.jdbc.Driver] at URL [jdbc:mysql://localhost:3306/accolite_hibernate]
Feb 27, 2021 11:32:54 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH0001001: Connection properties: {password=****, user=root}
Feb 27, 2021 11:32:54 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH0001003: Autocommit mode: false
Feb 27, 2021 11:32:54 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl$PooledConnections <init>
INFO: HHH000115: Hibernate connection pool size: 20 (min=1)
Feb 27, 2021 11:32:54 PM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.MySQL5Dialect
Feb 27, 2021 11:32:55 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH0001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConnectionAccess@6850b758] for
Hibernate: create table Categories (Category_Id integer not null, Category_Name varchar(255), Product_Product_Id integer, primary key (Category_Id)) en
Hibernate: create table Product (Product_Id integer not null, Product_Name varchar(255), primary key (Product_Id)) engine=MyISAM
Hibernate: create table Supplier (Supplier_Id integer not null, Supplier_Name varchar(255), primary key (Supplier_Id)) engine=MyISAM
Hibernate: alter table Categories add constraint FKknj3ng911bl186mtq9waq4q foreign key (product_Product_Id) references Product (Product_Id)
Hibernate: alter table Categories add constraint FKSpoe8yvtmuc2u4aw0h5q0lno foreign key (categories_Product_Id) references Product (Product_Id)
Hibernate: alter table Categories_Supplier add constraint FK1wh61y5712qct8dp8xmqg896c foreign key (supplier_Supplier_Id) references Supplier (Supplier_Id)
Hibernate: alter table Categories_Supplier add constraint FKtfce93pgi4cfyfgtbgsgio foreign key (categories_Category_Id) references Categories (Category_Id)
Feb 27, 2021 11:32:56 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
Hibernate: select categories_Category_Id, categories_Category_Name as category2_0, categories_Product_Product_Id as product_3_0 from Categories categories_ where categories_Cat
Hibernate: select supplier_Supplier_Id, supplier_Supplier_Name as supplier2_3 from Supplier supplier_ where supplier_Supplier_Id=?
Hibernate: select categories_Category_Id, categories_Category_Name as category2_0, categories_Product_Product_Id as product_3_0 from Categories categories_ where categories_Cat
Hibernate: select categories_Category_Id, categories_Category_Name as category2_0, categories_Product_Product_Id as product_3_0 from Categories categories_ where categories_Cat
Hibernate: select supplier_Supplier_Id, supplier_Supplier_Name as supplier2_3 from Supplier supplier_ where supplier_Supplier_Id=?
Hibernate: insert into Product (Product_Name, Product_Id) values (?, ?)
Hibernate: insert into Categories (Category_Name, product_Product_Id, Category_Id) values (?, ?, ?)
Hibernate: insert into Supplier (Supplier_Name, Supplier_Id) values (?, ?)
Hibernate: insert into Categories (Category_Name, product_Product_Id, Category_Id) values (?, ?, ?)
Hibernate: insert into Supplier (Supplier_Name, Supplier_Id) values (?, ?)
Hibernate: update Categories set categories_Product_Id=? where Category_Id=?
Hibernate: update Categories set categories_Product_Id=? where Category_Id=?
Hibernate: insert into Categories_Supplier (categories_Category_Id, supplier_Supplier_Id) values (?, ?)
Hibernate: insert into Categories_Supplier (categories_Category_Id, supplier_Supplier_Id) values (?, ?)
Hibernate: insert into Categories_Supplier (categories_Category_Id, supplier_Supplier_Id) values (?, ?)
Hibernate: insert into Categories_Supplier (categories_Category_Id, supplier_Supplier_Id) values (?, ?)
Hibernate: update Categories set categories_Product_Id=? where Category_Id=?
Hibernate: insert into Categories_Supplier (categories_Category_Id, supplier_Supplier_Id) values (?, ?)
-----Product to Categories : One to Many Done !!!!!-----
-----Categories to Products : Many to One Done !!!!!-----
-----Categories to Suppliers : Many to Many !!!!!-----
```

2. Product Table

The screenshot shows a SQL query editor with the query `SELECT * FROM Product;` and its results. The results pane displays a table with two columns: `Product_Id` and `Product_Name`. The data includes two rows: `101 Product 1` and `102 Product 2`.

Product_Id	Product_Name
101	Product 1
102	Product 2

3. Categories Table

The screenshot shows a SQL query editor with the query `SELECT * FROM categories;` and its results. The results pane displays a table with three columns: `Category_Id`, `Category_name`, and `product_Product_Id`. The data includes three rows: `1 Category 1` with `101`, `3 Category 3` with `101`, and `2 Category 2` with `102`.

Category_Id	Category_name	product_Product_Id
1	Category 1	101
3	Category 3	101
2	Category 2	102

4. Supplier Table

The screenshot shows a SQL query editor with the query `SELECT * FROM supplier;` and its results. The results pane displays a table with two columns: `Supplier_Id` and `Supplier_Name`. The data includes three rows: `999 Supplier 1`, `777 Supplier 3`, and `888 Supplier 2`.

Supplier_Id	Supplier_Name
999	Supplier 1
777	Supplier 3
888	Supplier 2

5. Categories_Supplier Table

Query +

Autocomplete: [Tab]->Next Tag. [Ctrl+Space]->List All Tags. [Ctrl+Enter]->List M

```
1 SELECT * FROM categories_supplier;
```

1 Result 2 Profiler 3 Messages 4 Table Data 5 Info

(Read Only)

<input type="checkbox"/>	Categories_Category_Id	supplier_Supplier_Id
<input type="checkbox"/>	1	777
<input type="checkbox"/>	1	999
<input type="checkbox"/>	2	888
<input type="checkbox"/>	3	777
<input type="checkbox"/>	3	999