

1. Create a stored procedure that takes a product id as input parameters and returns the details of the product.

```
DELIMITER $$
```

```
Create Procedure GetProductDetail(IN product_id_ INT)
```

```
BEGIN
```

```
Select Product_id, Product_name, Quantity,Price
```

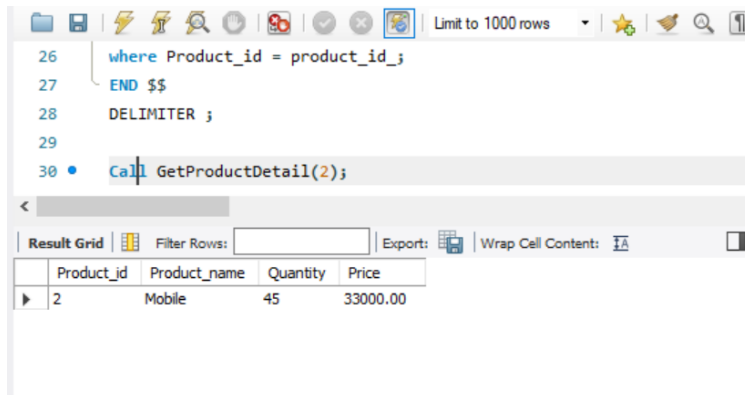
```
from Product_details
```

```
where Product_id = product_id_;
```

```
END $$
```

```
DELIMITER ;
```

```
Call GetProductDetail(2);
```

Example:**2. Create a function that takes price as an argument and return the details of the products whose price > argument.**

```
DELIMITER $$
```

Name: Gowtham Raja

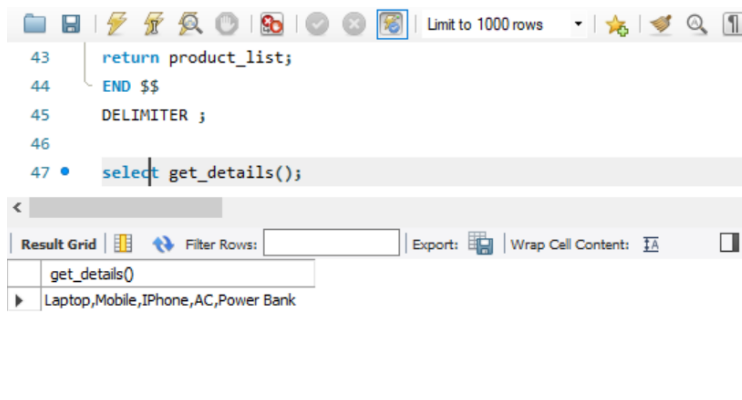
Date: 19-11-2024

Task: PLSQL

```
CREATE FUNCTION get_details()
returns varchar(255)
deterministic
begin
declare product_list varchar(255);
select group_concat(Product_name separator',') into product_list
from Product_details
where Price >1000;
return product_list;
END $$
DELIMITER ;

select get_details();
```

Example:



3.Return the entire table with stored procedure:

DELIMITER \$\$

Create Procedure GetEntireTable()

Name: Gowtham Raja

Task: PLSQL

Date: 19-11-2024

Begin

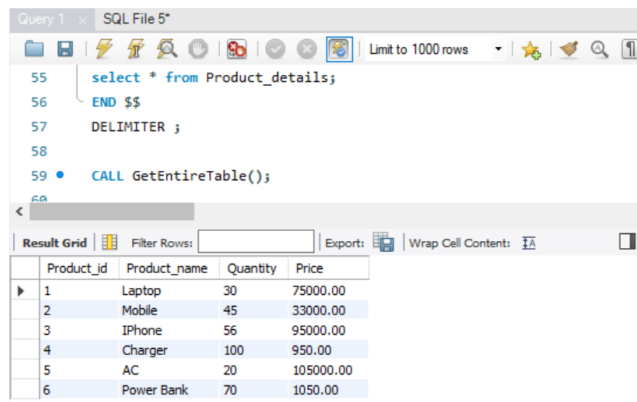
```
select * from Product_details;
```

```
END $$
```

```
DELIMITER ;
```

```
CALL GetEntireTable();
```

Example:



The screenshot shows a SQL IDE window titled "Query 1" and "SQL File 5". The query editor contains the following SQL code:

```
55 select * from Product_details;
56 END $$
57 DELIMITER ;
58
59 CALL GetEntireTable();
60
```

Below the query editor, the "Result Grid" is displayed, showing the results of the query. The grid has columns: Product_id, Product_name, Quantity, and Price. The results are as follows:

Product_id	Product_name	Quantity	Price
1	Laptop	30	75000.00
2	Mobile	45	33000.00
3	IPhone	56	95000.00
4	Charger	100	950.00
5	AC	20	105000.00
6	Power Bank	70	1050.00