Write a query to fetch all records from the movie table. Query must return all the columns in the result set.

```
select * from movie;

--
```

Display the details of column movie_name and cat from movie table. Provide aliasing to Column movie_name as name and cat as category as per the Expected Output.

```
select movie_name As name, cat As category from movie;
--
```

write a query to display profit done by the movies showing only movieid, movie_name and net_profit. net_profit to be calculated as the difference in productioncost and businesscost.

```
select
   movieid,
   movie_name,
   (businesscost - productioncost) As net_profit
from
   movie;

--
```

Write a query List the different categories of movies. Query must return only the list of unique categories as per the Expected Output shown below.

```
select distinct cat from movie;
--
```

Write a query to fetch movie_name,releasedate of the movie those belongs to Action Category.

```
select movie_name As movie_name, releasedate As releasedate from movie where cat = 'Action'
;
--
```

Write a sql query to fetch movie_name,productioncost from movies whose productioncost is greater than 100000.

```
select movie_name As movie_name, productioncost As productioncost from movie where producti
oncost > 100000;
--
```

write a sql query to fetch movie_name of those movies whoes category is Action and business cost is greater than 130000.

```
select movie_name As movie_name from movie where cat = 'Action' AND businesscost >= 130000;
```

```
--
```

Write a query to fetch records from movie table whose category is Action and release date is between 2016-01-01 and 2017-01-01. Query must return movie_name, cat and releasedate.

```
select movie_name As movie_name, cat As cat,releasedate As releasedate from movie where cat
='Action' AND releasedate Between '2016-01-01' AND '2017-01-01';

--
```

Write a query to sort the records of movie table in ascending order of movie_name. Query must return movie_name in the result set.

```
select movie_name As movie_name from movie Order By movie_name ASC;

--
```

write a query to sort the record in the descending order of production cost. Query must return movie_name and productioncost in the result set.

```
select movie_name As movie_name, productioncost As productioncost from movie Order By produ
ctioncost DESC;

--
```

Write a query to fetch movie_name, releasedate of all Movies in category Horror or Adventure in ascending order of releasedate.

```
select movie_name As movie_name, releasedate As releasedate from movie where cat In('Horror
','Adventure') Order By releasedate ASC;

--
```

write a query to fetch only first 3 records from movie table. Query must return movie_name, cat, releasedate in result set.

```
select movie_name As movie_name, cat As cat, releasedate As releasedate from movie LIMIT 3;

--
```

write a sql query to find the records whose movie_name starts with T. Query must return movie_name.

```
select movie_name from movie where movie_name like 'T%';
--
```

write a sql query to find the records whose movie_name ends with e. Query must return movie_name.

```
select movie_name from movie where movie_name Like '%e';
--
```

write a sql query to find the records whose movie_name contains ng at any position. Query must return movie_name.

```
select movie_name from movie where movie_name like '%ng%';

--
```

write a query to fetch a records that contain character n at third position in the movie_name. Query must return movie_name in the result set.

```
select movie_name from movie where movie_name like '__n%';

--
```

Write a query to find 2nd highest productioncost from the movie table. Query must return productioncost.

```
select Max(productioncost) As productioncost from movie where productioncost <(select Max(p
roductioncost) from movie);

--
```