

Hands on with Querying using Regex Expression

In order to perform following queries use a collection `products` with the following documents:

```
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }  
{ "_id" : 101, "sku" : "abc789", "description" : "First line\nSecond line" }  
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before   line" }  
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

1. find the documents whose description starts with "M".

```
> db.product.find({description:{$regex:/^M/}})
```

```
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }  
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

```
> db.product.find({description:{$regex:/^M/}})  
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }  
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

2. find the documents whose description end with "n"

```
> db.product.find({description:{$regex:/n$/}})
```

```
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

```
> db.product.find({description:{$regex:/n$/}})  
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }  
>
```

3. find the documents whose description contains "line" word"

```
> db.product.find({description:{$regex:/line/}})
```

```
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
```

```
{ "_id" : 101, "sku" : "abc789", "description" : "First line\nSecond line" }
```

```
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }
```

```
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

```
> db.product.find({description:{$regex:/line/}})
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
{ "_id" : 101, "sku" : "abc789", "description" : "First line\nSecond line" }
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

4. find the documents whose description contains second character "i".

```
> db.product.find({description:{$regex:/^i/}})
```

```
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
```

```
{ "_id" : 101, "sku" : "abc789", "description" : "First line\nSecond line" }
```

```
> db.product.find({description:{$regex:/^i/}})
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
{ "_id" : 101, "sku" : "abc789", "description" : "First line\nSecond line" }
```

5. find the documents where "sku" fields contains "xyz", ignore the case sensitivity.

```
> db.product.find({sku:{$regex:/xyz/i}})
```

```
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }
```

```
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

```
> db.product.find({sku:{$regex:/xyz/i}})
{ "_id" : 102, "sku" : "xyz456", "description" : "Many spaces before line" }
{ "_id" : 103, "sku" : "XYZ789", "description" : "Multiple\nline description" }
```

6. find the documents where any line from the description starts with 'S..

```
> db.product.find({description:{$regex:/^S/, $options:'si'}})
```

```
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
```

```
> db.product.find({description:{$regex:/^S/, $options:'si'}})
{ "_id" : 100, "sku" : "abc123", "description" : "Single line description." }
```