

### Query1

```
SELECT *  
FROM tls201_appln  
WHERE appln_auth= 'DK'
```

Remark: \* means “all columns from the table”

### Query2

```
SELECT *  
FROM tls209_appln_ipc  
WHERE ipc_class_symbol LIKE 'F03D%'
```

Remark: % is a wildcard often used in combination with LIKE, it will retrieve all IPC codes that start with F03D. F.ex.: F03D 1/04 , F03D 3/04,...

### Query3

```
SELECT *  
FROM tls206_person  
WHERE person_name LIKE 'vestas%'
```

Remark: each unique combination of a name + address + country is considered a new applicant or inventor. (Whatever detail it would be.)

### Query4

```
SELECT appln_id  
FROM tls201_appln  
WHERE appln_auth = 'IE'
```

Remark: each application in the database is defined by unique application identifier.

### Query5

```
SELECT appln_auth, appln_nr, appln_kind, appln_filing_date  
FROM tls201_appln  
WHERE tls201_appln.appln_auth = 'UY' ;
```

Remark: you can take any country you like, but keep in mind that large offices such as US, CN, EP, JP will result in millions of records being retrieved.

### Query6

```
SELECT *  
FROM tls201_appln  
WHERE (appln_auth= 'BR') or (appln_auth= 'MX') ;
```

Remark: brackets can be used in combination with operators so that the data sample can be extended or limited.

### Query7

```
SELECT person_name, person_address
FROM tls206_person
WHERE person_ctry_code = 'TR' ;
```

Remark: the person country code is the place of business (residence), not nationality.

### Query8

```
SELECT appln_title, appln_id
FROM tls202_appln_title
WHERE tls202_appln_title.appln_title LIKE '%hybrid%motor%';
```

Remark: LIKE is mostly used in combination with the % wildcard.

### Query9

```
SELECT appln_auth, appln_nr, appln_kind, appln_filing_date
FROM tls201_appln
WHERE (appln_auth = 'AT' ) AND (appln_kind = 'W')
      AND (appln_filing_date> '2005-01-01' );
```

Remark: PCT applications are identified through the application kind code = 'w', the application authority is the country where the PCT was filed. PCT applications filed in Geneva have country code = 'IB' for International Bureau.

### Query10

```
SELECT tls224_appln_cpc.appln_id
FROM tls224_appln_cpc
WHERE tls224_appln_cpc.cpc_class_symbol LIKE 'Y02E 10/7%'
```

Remark: duplicates can occur because an application can have multiple CPC codes starting with Y02E10/7...

### Query11

```
SELECT tls201_appln.appln_auth, tls201_appln.appln_nr, tls201_appln.appln_kind,
tls209_appln_ipc.ipc_class_symbol
FROM tls201_appln JOIN tls209_appln_ipc ON tls201_appln.appln_id =
tls209_appln_ipc.appln_id
WHERE tls201_appln.appln_auth ='US' and appln_filing_date between '2015-04-01'
and '2015-04-30'
order by tls201_appln.appln_id, ipc_class_symbol;
```

Remark: JOIN is used to merge tables, the ON clause specifies the condition(s) for the join. The “order by” sorts the result list.

### Query11

```
SELECT tls201_appln.appln_auth, tls201_appln.appln_nr, tls201_appln.appln_kind,  
tls206_person.person_name  
FROM (tls201_appln INNER JOIN tls207_pers_appln ON tls201_appln.appln_id =  
tls207_pers_appln.appln_id) INNER JOIN tls206_person ON  
tls207_pers_appln.person_id = tls206_person.person_id  
WHERE tls201_appln.appln_auth = 'EP' AND tls206_person.person_name like  
'%shell%' and applt_seq_nr > 0;
```

Remark: using wild cards in names can lead to unexpected results with ambiguous names. It is wise to check first what names are effectively selected, and adapt the query to exclude unwanted applicants. The “applt\_seq\_nr > 0” condition forces the database too only select applicants and not inventors, which are in the same tls206\_person table.

### Query12

```
SELECT tls201_appln.appln_auth, tls201_appln.appln_nr, tls201_appln.appln_kind,  
tls206_person.person_name  
FROM (tls201_appln INNER JOIN tls207_pers_appln ON tls201_appln.appln_id =  
tls207_pers_appln.appln_id) INNER JOIN tls206_person ON  
tls207_pers_appln.person_id = tls206_person.person_id  
WHERE tls201_appln.appln_auth = 'EP' AND tls206_person.person_name like  
'%shell%' and applt_seq_nr > 0;
```

Remark: using wild cards in names can lead to unexpected results with ambiguous names. It is wise to check first what names are effectively selected, and adapt the query to exclude unwanted applicants. The “applt\_seq\_nr > 0” condition forces the database too only select applicants and not inventors, which are in the same tls206\_person table.