

## CONSULTAS:

1. Buscar las solicitudes que estén prontas a vencer (menos de 3 días)

```
SELECT s.idSolicitud as 'Solicitudes Próximas a Vencer',  
p.fechaInicio, p.fechaLimite, p.fechaLimite-current_date() as  
'Días restantes'
```

```
FROM solicitud s
```

```
JOIN PAGO P USING (idSolicitud)
```

```
WHERE p.estadoDePago='Por Pagar'
```

```
AND p.fechaLimite-current_date()BETWEEN 0 AND 3;
```

2. Buscar el estudiante que más solicitudes ha realizado

```
SELECT u.idUsuario,u.nombre, u.apellido, u.correoElectronico,  
t.cant AS 'Cantidad Máxima'
```

```
FROM usuario u
```

```
join (SELECT idUsuario , COUNT(*) as cant
```

```
FROM solicitud s
```

```
JOIN usuario u USING(idUsuario)
```

```
WHERE u.tipo = 'estudiante'
```

```
GROUP BY (idUsuario)) as t
```

```
using(idUsuario) WHERE t.cant=(
```

```
SELECT MAX(cant) FROM (select COUNT(*) as cant
```

```
FROM solicitud s
```

```
JOIN usuario u USING(idUsuario)
```

```
WHERE u.tipo = 'estudiante'
```

```
GROUP BY (idUsuario)) as t
```

```
);
```

### 3. Buscar el trámite más solicitado

```
SELECT tr.idTramite as 'Trámite más  
solicitado',tr.nombre,tr.descripcion,t.cant as 'Cantidad Máxima'  
from tramite tr  
JOIN (SELECT idTramite,COUNT(*) as cant  
FROM solicitud  
JOIN tramite USING(idtramite)  
GROUP BY(idTramite)) as t using(idTramite)  
WHERE t.cant=(SELECT MAX(cant)  
FROM (SELECT idTramite,COUNT(*) as cant  
FROM solicitud  
JOIN tramite USING(idtramite)  
GROUP BY(idTramite)  
) AS t);
```

### 4. Calcular el promedio de tiempo que tarda cada usuario en completar una solicitud:

```
SELECT u.idUsuario,u.nombre,u.apellido,u.correoElectronico,  
CONCAT(CAST(ROUND((Sum(p.fechaDeCancelacion)-  
SUM(p.fechaInicio))/COUNT(*)) AS CHAR),' Días') as 'Días en pagar  
un trámite'  
From pago p  
JOIN solicitud s USING(idSolicitud)  
JOIN usuario u USING(idUsuario)  
Where p.fechaInicio IS NOT NULL  
AND estadoDePago='Pagado'  
GROUP BY(idUsuario);
```

5. Probabilidad de que un tramite quede en  
(Completado,enProceso,pendiente)

```
SELECT estado,CONCAT(CAST(ROUND(((t2.cant/total)*100),2) AS  
CHAR),'%') as 'Proporción de tramites' FROM(  
SELECT SUM(cant) as total FROM(  
SELECT ESTADO,COUNT(*) as cant  
FROM SOLICITUD  
JOIN USUARIO USING(IDUSUARIO)  
group by(estado))AS t) as t1,  
(SELECT ESTADO,COUNT(*) as cant  
FROM solicitud  
JOIN USUARIO USING(IDUSUARIO)  
group by(estado)  
) AS t2;
```

6. Para el número de solicitudes de cada trámite por cada estado

```
SELECT t.nombre AS Solicitud, s.estado AS Estado,  
COUNT(s.idSolicitud) AS Cantidad  
FROM tramite AS t  
INNER JOIN solicitud AS s USING(idTramite)  
GROUP BY t.nombre, s.estado;
```

7. Promedio de costo y número de trámites por unidad

```
SELECT u.nombreUnidad AS Unidad, round(AVG(t.costo), 2) AS Costo,  
COUNT(u.idUnidad) AS CantidadTramites  
FROM tramite AS t INNER JOIN unidad AS u USING(idUnidad)  
GROUP BY idUnidad;
```

#### 8. Funcionarios por número de solicitudes que gestiona

```
SELECT s.idFuncionario, u.nombre, u.apellido,  
CONCAT(CAST(COUNT(s.idFuncionario) AS CHAR), " solicitudes") AS  
Gestiona  
  
FROM solicitud AS s INNER JOIN usuario AS u ON s.idFuncionario =  
u.idUsuario GROUP BY s.idFuncionario;
```

#### 9. Cantidad de solicitudes por tipo de usuario

```
(SELECT u.tipo, COUNT(*) AS Cantidad  
FROM solicitud s  
INNER JOIN usuario u ON s.idUsuario = u.idUsuario  
GROUP BY u.tipo)  
UNION  
(SELECT 'Total', COUNT(idSolicitud) FROM solicitud);
```

#### 10. Número de solicitudes en proceso por cada usuario no administrativo

```
SELECT nombre, apellido,  
(SELECT COUNT(*)  
FROM solicitud AS s  
WHERE s.idUsuario = u.idUsuario  
AND s.estado = 'en proceso') AS 'Solicitudes en Proceso'  
FROM usuario AS u WHERE u.tipo
```