

Caso N1 Implementación de Bloques PL/Sql

```
Create Table Avion(  
Piso number not null,  
Asiento number not null,  
Clase number not null,  
Disponible char not null,  
Constraint Tabla_Avion primary key (piso,asiento,clase));
```

```
Declare  
  
P_Piso Avion.Piso%TYPE,  
P_Asiento Avion.Asiento%TYPE,  
P_Clase Avion.Clase%TYPE,  
P_Disponible Avion.Disponible%TYPE := 'S';
```

```
Begin  
  
For P_Piso IN 1..3 LOOP  
  
If P_Piso = 1 Then  
  
For P_Asiento IN 1..216 LOOP  
  
P_Dis := '& P_Dis';  
  
Insert into Avion(Piso,Asiento,Clase,Disponible)  
  
Values (P_Piso, P_P_Asiento,4,P_Dis);  
  
End LOOP;  
  
Elseif p_Piso =2 then  
  
For p_Asiento IN 217..236 LOOP  
  
P_Dis:= '& P_Dis';  
  
Insert into Avion(Piso,Asiento,Clase,Disponible)  
  
Values (P_Piso, P_P_Asiento,2,P_Dis);  
  
End LOOP;
```

```

For P_Asiento IN 237..277 LOOP
P_Dispatch := '& P_Dispatch';
Insert into Avion(Piso,Asiento,Clase,Disponible)
Values (P_Piso, P_P_Asiento,3,P_Dispatch);
End LOOP;

Else
For P_Asiento IN 278..288 LOOP
P_Dispatch := '& P_Dispatch';
Insert into Avion(Piso,Asiento,Clase,Disponible)
Values (P_Piso, P_P_Asiento,1,P_Dispatch);
End loop;
End IF;
End Loop;
End;

```

Prob2

a) Crear objetos de datos.

```

CREATE TABLE Temporadas(
nombre varchar(10) not null,
mesinicio int not null,
diainicio int not null,
mesfinalizacion int not null,
día_finalizacion int not null,
constraint reservas_pk primary key (nombre, mesinicio, diainicio));

```

```
CREATE TABLE estadistichotel (  
  Id_cadena number not null,  
  Nombre varchar2(50),  
  Habitacionesocupadas number,  
  Habitacionesdisponibles number,  
  Habitacionesreservadas number,  
  Constraint id_cadena_fk primary key (id_cadena));
```

```
CREATE TABLE Habitaciones(  
  Id_cadena number,  
  Piso int not null,  
  Habitacion varchar(2) not null,  
  Tipo varchar(10) DEFAULT 'Doble',  
  Constraint Tipo_not_null Tipo not null,  
  Constraint chk_Tipo CHECK( Tipo = 'Individual' Tipo = 'Doble' Tipo = 'Suite'),  
  Constraint pk_habitaciones primary key(Piso, Habitacion),  
  constraint fk_habitaciones foreign key(Id_cadena), references estadistichotel(Id_cadena));
```

```
CREATE TABLE Reservas(  
  piso int,  
  habitacion VARCHAR2(2) not null,  
  fechaentrada date,  
  fecha DEFAULT sysdate,  
  noches int,  
  primary key (piso, habitacion, fechaentrada),  
  constraint fk_reservas_piso foreign key (piso) references habitaciones (piso),  
  constraint fk_reservas_habitacion foreign key (habitacion) references habitaciones (habitacion));
```

b) Procedimiento para insertar en la tabla temporadas.

```
Create or replace procedure Crear_temporadas(
P_nombre Temporadas.nombre%TYPE,
P_mesinicio Temporadas.mesinicio%TYPE,
P_diainicio Temporadas.diainicio%TYPE,
P_mesfin Temporadas.mesfinalizacion%TYPE,
P_diafin Temporadas.dia_finalizacion%TYPE) AS
Begin
Insert Into (nombre, mesinicio, diainicio, mesfinalizacion, dia_finalizacion)
Values (P_nombre, P_mesinicio, P_diainicio, P_mesfin, P_diafin);
Exception
When DUP_VAL_ON_INDEX Then
DBMS.OUTPUT.PUT_LINE ('dato existente');
When OTHERS THEN DBMS.OUTPUT.PUT_LINE ('No se existe el registro');
COMMIT;
End Crear_temporadas;

Begin
Crear_temporadas('Alta',6,1,8,30);
Crear_temporadas('Medio',1,1,31,5);
Crear_temporadas('Baja',1,9,31,12);
End;

Create or replace procedure crear_habitacion(
P_ID_Cadena Habitaciones.ID_Cadena%Type,
P_Piso Habitaciones.Piso%Type,
P_Hab habitaciones.Hab%Type,
P_Tipo habitaciones.Tipo%Type) AS
```

```

BEGIN

FOR P_Piso IN 1..12 LOOP

IF P_Piso >=1 AND P_Piso<=11

FOR x IN 1..25 LOOP

IF P_Habitacion = 20 THEN

P_Tipo := 'Dobles';

ELSE

P_Tipo := 'Sencilla';

END IF;

INSERT INTO habitaciones(piso, Habitacion, Tipo)

VALUES(P_Piso,x, P_Tipo);

END LOOP;

ELSE

FOR y IN 1..8 LOOP

P_Tipo := 'Suites';

INSERT INTO habitaciones ( piso, Habitación, Tipo)

VALUES(P_Piso,y, P_Tipo);

END LOOP;

END IF;

END crear_habitacion;


CREATE OR REPLACE FUNCTION habitac_o (
Habitacion habitaciones.habitacion%TYPE,
Habitacionesdisponibles estadisticashotel.habitacionesdisponibles%TYPE)
RETURN NUMBER AS

Begin

Return habitaciones - habitacionesdisponibles;

End habitac_o;

```

```
CREATE OR REPLACE FUNCTION habitac_d (  
Habitación habitaciones.habitacion%TYPE,  
Habitacionesocupadas estadisticashotel.habitacionesocupadas%TYPE)  
RETURN NUMBER AS  
  
Begin  
  
Return habitaciones - habitacionesocupadas;  
  
End habitac_d;
```