

DBak atzitzeko aplikazioak ARIKETAK

IRAKASLEAK (nhan, izena, titula, ..., bere-saila)

IRAKASGAIK (kodea, izena, kredituak, ..., bere-saila)

PROJEKTUA (kodea, izena, aurrekontua, ..., bere-saila)

IKASLEA (izan, izena, helbidea, ...)

IKAS-IRAKAS (inan, kodea, nota)

Aurreko domeinua kontuan hartuz:

1. Irakasaiaren izena eskuatu eta jarraian bere saila, irakasai hori hartu duten (berian matrikulatuta duden) ikasleen izen-zerenada eta ikasle-kopurua pantailatzan dituen programa diseina ezazu.
2. Gorburukoan sailaren izena dagoela, sail bakoitzaren irakaslaak, irakassalak eta projektuak erakusiko dituen istitua lortzen duen programa diseina ezazu.
3. Hamar ikasle orienten izena lortzen duen programa diseina ezazu, hots, batezbesteko nota onena duten ikasleen izena. Ikasle batzuk matrikulatu berriak izango direla eta beraz bere batezbesteko nota null izango dela kontuan izan.

BEZEROA (inan, bizen, hiria, beherap)

SALTZALEA (sنان, sizen, hiria, portzenttua)

PRODUKTUA (koda, pizan, hiria, kopurua, prezioa)

ESKAERA (eskzenb, hilabetea, bhan, sman, kodea, kanitatea, ordainzeko)

Aurreko domeinua kontuan hartuz:

4. Karera amaterako projektuen inguruiko informazioa bildu nahi da. Lehen fasean ikaslek gustukoien dituzten hiru proiektuen izena eman beharko dute, aztienean batzuk geldizteko. Arriketa honetan lehen fasea hobetan zentratutako gara. Horretarako honako E-E diagrama eta bere taulak kontuan hartuko ditugu:

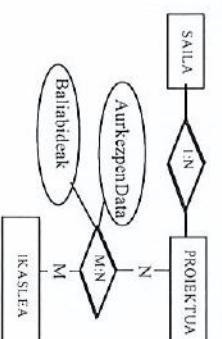


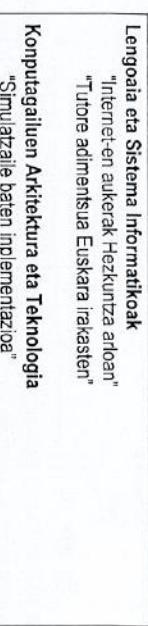
Diagrama horretatik honako taulak lortzen dira:
Hiria (izen, bizi-maila, historia, paisaien interesa, hondartzak/kalitatea), Ostalua (izen, bizi-maila, historia, gela-kop, hiria, kategokodea), Bezeroa (inan, bizen, helbidea, VISA-zenbaki), Kategorial (kodea, prezioMax, prezioMin), Erreserba (izen, han, data).

Hiria zentzutako erreserba sailkatzen dira: bizi-maila, bere historia, paisaien interesa, hondartzak/kalitatea, Ezagarrirako bakoitzak 0 eta 10 bitarteko balio bat esleitzen zaio. Hiria bakoitzean ostalua desberdinak daude, bakoitzera bere kategoria osoan. Ostaluetan erreserba egiten dira eta bezero batek ostatu desberdinietan erreserba eginda edukitzeara geratzen da. Sistemaak bezeroan ostaluetan erreserba egiteko aukera emango dio, honek aukeratutako hirietan dsuden ostaluetan. Hirien aukera bere historiari, bizi-maila, etabarren arabera egingo du. Sisteman pauso hausk jarraituko dira:

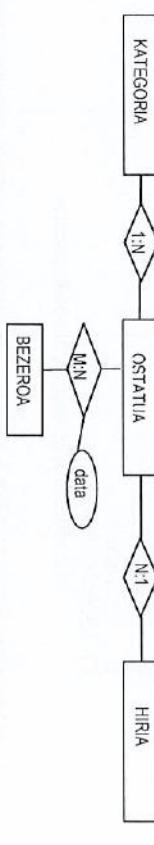
1. Hiria bakoitzeko erabiltzen den ezagarrirako sisteman bezerrari balio bat eskatuko dio. Balio hori 0 eta 10 bitarteko izango da. Adibidez, honen antzeko pantaila bat aterako da:
- Honako ezagarririetan zure hiri idealek bete beharko litzkeen baliokoak sartu, mesedez. Balioa 0 (ez du garrantziak) eta 10 (garantza dago) bitartekoak izan behar da:

bizi-maila:
Historia:
Paisaien interesa:
hondartzak/kalitatea:

1. Ikslearren NAN zenbakia eskuatu eta ikase horrek projektuen aukeraketarik ez duela egin egiaz-tatu ondoren
2. oraindik hautagai diren projektuen zerrenda erakusten du, hots, 5 ikaslek baino gutxiagok hartu duten projektuen zerrenda. Projektu horiek sailaren anabera banatuta agertuko dira, indiar erakusten den moduan:



5. Informazio turistikoa eskainiko duen sistema eraiki nahi da. Sistema honetan ditugun datuak honako eskeinaren arabera antolatu daude:



6. Matrikula taularen gainean integritate-muritzapen bat definituta dago. Muritzapen horrek projektu bat aukeratu duten ikasleen kopurua 5 baino handiagoa ez dela egiazlatzen du. Muritzapena ez bada baleztzen, sistemak salbuksesen bat sortzen du, bere kodea "CONSTRAINT_VIOLATED" delarik (`e.getErrorCode()=CONSTRAINT_VIOLATED`). Gainera kontuan hartu behar dugu: a) Projektuak eskaintzen EZ duten sailak egin daitezkeela; eta b) ikasle bakoitzak 3 projektu aukeratzeko dituelarik.

Honakoa egiten duen eta SQL erabilizten duen programa idatz ezazu:

```

CREATE TABLE IKASLEAK (
    ID INT PRIMARY KEY,
    NOMBRE VARCHAR(50),
    APELLIDOS VARCHAR(100),
    DNI VARCHAR(9),
    FECHA_NACIMIENTO DATE,
    DIRECCION VARCHAR(100),
    LOCALIDAD VARCHAR(50),
    CODIGO_POSTAL INT,
    PROVINCIA VARCHAR(50),
    TIPO_SALIDA ENUM('Estudiante', 'Profesional'),
    ESTADO ENUM('Activo', 'Inactivo')
);

CREATE TABLE PROYECTOS (
    ID INT PRIMARY KEY,
    TITULO VARCHAR(100),
    AREA_BALIZONADO VARCHAR(50),
    RESPONSABLE_ID INT,
    ESTADO ENUM('Abierto', 'Cerrado')
);

CREATE TABLE SALIDAS (
    ID INT PRIMARY KEY,
    TIPO_SALIDA VARCHAR(50),
    DIA DATE,
    HORARIO TIME,
    LUGAR VARCHAR(100)
);

CREATE TABLE ESTADISTICAS (
    ID INT PRIMARY KEY,
    PROYECTO_ID INT,
    IKASLE_ID INT,
    SALIDA_ID INT,
    CANTIDAD INT
);

CREATE TABLE BEZEROA (
    ID INT PRIMARY KEY,
    NOMBRE VARCHAR(50),
    DIRECCION VARCHAR(100),
    LOCALIDAD VARCHAR(50),
    CODIGO_POSTAL INT,
    PROVINCIA VARCHAR(50),
    TIPO_BIZIBALIO ENUM('Residencial', 'Comercial', 'Servicio')
);

CREATE TABLE HIRIA (
    ID INT PRIMARY KEY,
    NOMBRE VARCHAR(50),
    DIRECCION VARCHAR(100),
    LOCALIDAD VARCHAR(50),
    CODIGO_POSTAL INT,
    PROVINCIA VARCHAR(50),
    TIPO_HIRIA ENUM('Residencial', 'Comercial', 'Servicio')
);

CREATE TABLE OSTALUA (
    ID INT PRIMARY KEY,
    KATEGORIA_ID INT,
    HIRIA_ID INT,
    PREZIO_MAX DECIMAL(10,2),
    PREZIO_MIN DECIMAL(10,2)
);

CREATE TABLE KATEGORIA (
    ID INT PRIMARY KEY,
    NOMBRE VARCHAR(50),
    DESCRIPCION TEXT
);

CREATE TABLE ERRESERBA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_VISA_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_HISTORICO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_ERRESERBA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_OSTALUA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(10,2)
);

CREATE TABLE SALIDA_PAISETA_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO_VISA_BIZIBALIO (
    ID INT PRIMARY KEY,
    HIRIA_ID INT,
    KATEGORIA_ID INT,
    FECHA DATE,
    HORA TIME,
    DURACION INTERVAL DAY TO SECOND,
    PREZIO DECIMAL(1
```

2.- Esandako baloak betezen dituzten hirien zerrenda itzuliko du sistemak, hau da, ezagutu guzie-
tan bezeroak esandako baino balio handiago bat duten hirak. Horrela, sistemak bigarren pantaila
bat aterako du, hirien zerrenda rekin, bat aukeratzeko eskatuz. Adibidez:

Zuk esandako baloak betetzen dituzten hirak honakoak dira:
Donostia
Santander
Gijón
A Coruña

Bat aukera ezazu:

3.- Sistemak aukeratutako hiriañen ostatuen zerrenda itzuliko du. Zerrenda horretan ostatuak katego-
riaren arabera sailkatuta azaltuko dira, eta maila zehatz batean ostatuak EZ badago, mezu egoki
bat aterako da. Adibidez:

(1) DONOSTIA
BOSTIZARREKO HOTELAK
Maria Cristina
LAU IZARREKO HOTELAK
Costa Vasca
Aranzazu-Donostia
Londres
HIRU IZARREKO HOTELAK
Amara Plaza
Europa
BIIZARREKO HOTELAK
Hiri horietan ez dago kategoria horretako ostatuak
...

4.- Ondoren sistemak honako datuak eskatuko ditu: bezeroaren nan zenbakia, nahi den ostatuaren
izena, sarrera-data eta egonaldiaren egun-kopurua.

5.- Sistemak egun horretarako plaza libreak daudela egiazatzeko du, eta balekoan erreserba egingo
du. Adibidez, sarrera-data 99/12/19 izanik egonaldia 3 egunetako bada, sistemak erreserba 99/12/9,
99/12/10 eta 99/12/11 egunetako egingo du. Plaza libreak egun GUZTIETARAKO ez badaude
ez da erreserbarik egingo. Data batean plaza librenik egingo da egun horretarako egindako erre-
serben kopurua hautatutako ostate horrek dituen plazen kopunia (Ostatua taularen gelabekop atr-
butua) baino txikiagoa denean.

6.- Erreserba ezingo balitz egin, bezeroari dagokion mezuaz alera eta 4. pausora itzuliko lizateke
(erreserba egiteko, gehienez ere, beste bi aukera emanez)

1. Arbeit

try {

:

#sql iterB1 = {select NA, ^{Tfnotebilie} from Hauptsaldo_Bezero INNER JOIN
Bezero ON HTB.NA = Bezero.NA where HTB.Helmoga = 'Bilbo'
as HTB}

HTB.intereDate = '2012-05-31' and HTB.Zenbilie = 5};

System.out.println("NA " + ^{and Helmoga = 'Bilbo'} interDate);
while (iterB1.next()) {

try {

nan = iterB1.NA();

telefono = iterB1.Tfnotebilie();

System.out.println("Konto " + telefono);

if (telefono == goldeln(telefono)) {

#sql iterB2 = {select NA from Bezero where NA = :nan};

else { #sql iterB2 = {select NA from Bezero where NA = :nan};

#sql {set transaction isolation level read commit};

#sql {Delete from Hauptsaldo_Bezero where NA = :nan};

#sql {commit};

catch (SQLException e) {

#sql {rollback};

if (e.getErrorCode == "Constraint Violated") {

System.out.println("Doppelte bornule Bezeen frangolko bezero hori");

~~break~~ continue;

8

■

#sql {set transaction isolation level read committed};
#sql {insert into Hauptsaldo_Bezero values ('Bilbo', '2012-05-31', 12, Gestrid, 9:00, La_Casilla, 60)};
#sql {lock table Hauptsaldo_Bezero in shared mode};
while (iterB2.next()) {
nan = iterB2.NA();
#sql {UPDATE Hauptsaldo_Bezero SET Zenbilie = 12 WHERE Helmoga = Bilbo
and interDate = '2012-05-31' and Zenbilie = 5 and NA = :nan};
while (iterH1.next()) {
nan = iterH1.NA();
#sql {Delete from Hauptsaldo_Bezero WHERE Helmoga = Bilbo and interDate = '2012-05-31'
and Zenbilie = 5 and NA = :nan};

```
#Sql {Delete from Haldelextraage where Helmige = Bilbo  
and isterdat = 2012-05-21  
and zubelwe = 54;  
#sql kommt h
```

g

```
{catch (SQLException e) {  
    System.out.println ("Error: " + e.getErrorCode());  
    #Sql {rollBack};  
    exit(-1);
```

```
{catch (Exception e) {  
    System.out.println ("Error: " + e.getMessage());  
    exit(1);  
g
```

5. Anilka

```
import java.io.*;
import java.sql.*;
import java.util.*;
import javax.sql.DataSource;
#sql iterador IteradorBis (String linea);
#sql iterador IteradorBisBi (String linea, int linea);

class informacionBis {
    public static void main (String args[]) throws Exception {
        int bimail, Historia, paisaje, handkredit, Gkrip, gelaki;
        IteradorBis iterBis; iterBis;
        String linea1, linea2, han;
        Date date;
        BufferedReader sarrera = new BufferedReader (new InputStreamReader (System.in));
        String Driver = "Oracle.jdbc.driver.OracleDriver";
        String DBURL = "jdbc:oracle:thin:@192.168.0.10:1512:Erreka";
        String UJD = "Erreka";
        String PWD = "pasadita";
        try {
            System.out.println ("Hasko eraztunetan zure hiri idealek bete beharko
                liturakuen baliabek sar itzazu, mescob. Balioa 0 (ez du garrautik) eta 10 (halitzeak
                de) bitartekoak (zen behar da");
            System.out.println ("    bizi-maila: "); maila = sarrera.
                bimail = sarrera.nextInt();
            System.out.println ("    Historia: "); historia = sarrera.
                historia = sarrera.nextInt();
            System.out.println ("    Paisaje interes: "); paisaje = sarrera.
                paisaje = sarrera.nextInt();
            System.out.println ("    Handkredit kreditatea: "); handkredit = sarrera.
                handkredit = sarrera.nextInt();
            #sql iterBis = #Select linea from hiri where Bizi-maila >=:bimail and
                interesHistorialoc >= :historia and PaisajeInteres >= :paisaje and
                HandkreditKreditatea >= :handkredit;
```

System.out.println("Zile escondedas betreuen diktator hirte
hundert dira");

while (iterBt.next()) {

Izene = iterBt.getStr();

System.out.println(" " + Izene);

} iterBt.close();

System.out.println("Bat cultura c3odur")

Izene = Serre.readLine();

int i = 5;

~~Asq1 iterBt = {Select Oizene, libetekde from Ostetve where hirte = :Izene and libetekde >= 5};
while (i > 0) {~~

~~Asq1 iterBi = {Select Oizene, from Ostetve where hirte = :Izene and libetekde >= 5};~~

System.out.println(" " + "Oizene, libetekde from Ostetve where hirte = :Izene");

iS(iterBi.next());

while (iterBi.next()) {

Oizene = iterBi.getStr();

System.out.println(" " + Oizene);

}

else {

System.out.println(" Hiru kontua es dago libetekde kostekileti");

}

else {

while (true) {

System.out.println("Sortu van Beabidi");

no = Serre.readLine();

System.out.println("Sortu uhi den ostetveren (izene)");

Oizene = Serre.readLine();

System.out.println("Serre-data:");

data = Serre.readLine();

System.out.println("Egan lepuru sortu");

Glop = Serre.nextInt();

~~Asq1 {Select getlop into getlop; from Ostetve where Oizene = :Oizene};~~

for (i = 0; i < Glop; i++) {

~~Asq1 {Update carburante into :elop from Esreserbe where Oizene = :Oizene and~~

~~libre = 1; elop <= getlop and libre <= data; data = data + 1};~~

if (not libre) ... // berria seidu. continue;

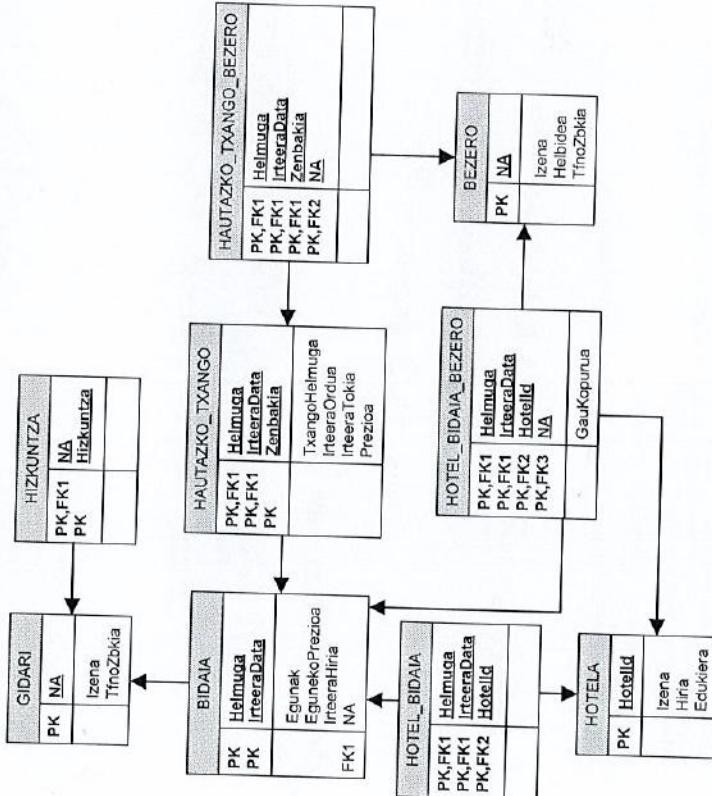
~~Asq1 {transaction isolation level read committed};~~

book for (i = 0; i < gwid; i++) { Asq1 insert into Esreserbe values (:no, :Oizene, :data); }

Datu-baseak. Programa SQLJn

Datu-baseak. Programa SQLJn

Bedi honako UML diagrama:



Gidari (NA, Izena, TinoZbikia)
Hizkuntza (NA, Hizkuntza)

Bidaia (Helmuga, InterraData, EgunkoPrezioa, InteraHiria, NA)
Hotela (HotelId, Izena, Hiria, Edukierra)
Hotel_Bidaia (Helmuga, InterraData, HotelId)
Bezero (NA, Izena, Helmidea, TinoZbikia)
Hotel_Bidaia_Bezero (Helmuga, InterraData, HotelId, NA, GauKopurua)
Hautazko_Txango (Helmuga, InterraData, Zentzia, TxangoHelmuga, InterraOrdua, InteraTokia, Prezioa)
Hautazko_Bezero (Helmuga, InterraData, Zentzia, NA)

Lanestik erretiratuta dauden pertsonentzat egun bat baino gehiago irauten duten bidaiaik antolatzen dituen ONDO-PASA empresak, datuak kudeatuko dituen aplikazio informatiko bat garatu du.

- Bidaia bat (adibidez, Benidorm-era) data desberdinetan egin daitake. Bidaia bakoitzak helmuga eta interra-data erabilizten dira. Gainera, bidaiaiaren iraupena egunetan, eguneko prezioa, interra-hiria, erabilgarri dauden hotelak eta bidaiai esleitutako gidari bakarra gordeko dira.
- Gidari bakoitzak bere NArekin identifikatzeko zenbat gau pasa ditzake, telefono-zenbakia eta hitz egiten dituen hizkuntzak ezagutzenean dira.
- Bezeroen, NArekin identifikatuak, izena, helbidea eta telefono-zenbakia ezagutzen dira.
- Hotel bakoitzeko bere identifikazio zenbakia bakarra, izena, hiria eta edukierra gordetako dira.
- Bidaia batean parte harzen duen bezero bakoitzeko zenbat gau pasa ditzuten bezeroak bidaia horri esleitutako hoteletan gordetzen nahi da. Bidaiai egun bat baino gehiago irauten duenek gerta daitake bidaia berean hotel bat baino gehiago esleitu izana.
- Bidaia bakoitzak hautazko txango bat edo gehiago eduki ditzake. Hautazko txango teera-tokia gordetako dira, (1, 2, 3, ...), helmuga, prezioa, interra-ordua eta irteera-tokia gordetako dira. Zein bezerok egin duen hautazko txango bakoitzak ere gordetako da.

Gasteizko Antium museoaan desinfekzio-tratamendu bat egingo da 2012-05-31an. Horren ondorioz, ezin izango da egun horretarako aurreikusita zegoen eta (Bilbo, 2012-05-31, 5) identifikazioa duen hautazko txangoa. Hautazko txango honek datu-basetik desageritzera doan txangoan zena emanda zeukanen bezeroei. Txango berria ez dago datubasean gordeta: (Bilbo, 2012-05-31, 12, Gasteiz, 9:00, La Casilla, 60). "Hautazko txangoan izen-ematea eguneratu/ezabatu" erabilpen-kasua implementatu duen Java programa idatz ezazu (2 puntu).

- (Bilbo, 2012-05-31, 5) hautazko txangoan izena eman duten bezeroak zein diren identifikatu:

NA	Telefonoa
1123	943566677
2234	688678901
3455	943500890

- Identifikatutako bezero bakoitzari telefonoa erabiliz gaidetu ea hautazko txango berrian izena eman nahi duen ala ez (erantzun posibleak: "bai", "ez") eta datu-basean aldaketa guztiek jasotzeko behar diren ekintzak gauzatu.

Oharra: Kontuan izan une berean eta bulego desberdinietatik sistemaren gaineko atzipoen konkurrentea egin ahal dela. Hori horrela izanik posible da, adibidez, hautazko txangoen izen-emateko prozesua hainbat erabiliztailek aldi berean egitea.
zena:

Datu-baseak. Programa SQLJn

```
import java.io.*;
import java.sql.*;
import sql.runtime.*;
import sql.runtime.ref.DefaultContext;
class ondopasa
{
    public static void main(String args[]) throws Exception
    {
        BufferedReader sarrera = new BufferedReader(new InputStreamReader(System.in));
        String DRIVER = "oracle.jdbc.driver.oracleDriver";
        String DBURL = "jdbc:oracle:thin:@000100.gi.en.es:1512:Ereala";
        String UID = "GRANITZALEA";
        String PWD = "PRAHITZA";
        try
        {
            /* Datu-basearekin konekzioa sortu eta konexio-testuinguru lehenetsia definitu */
            Driver controlador = (Driver) Class.forName(DRIVER).newInstance();
            DriverManager.registerDriver(controlador);
            DefaultContext test = new DefaultContext(DBURL, UID, PWD, false);
            DefaultContext.setDefaultContext(test);
            System.out.println("Konekzioa eginda");
            System.err.println("Errorea konekzioa: " + e.getMessage());
            System.exit(1);
        }
        catch (Exception e)
        {
            System.out.println("Errorea konekzioa: " + e.getMessage());
        }
    }
}
```

```
#SQL {Delete from Halkozko_txango where Helmug = "Bilbo"
      AND TtxedoDate = "2012-05-31" AND IdentiLizsio = "5";}
      #SQL {insert into Halkozko_txango values ('Bilbo', 2012-05-31,
      12, Gasteiz, 9.00, 'Le Castille', '60')};
      while (IterBi.next()) { nom = IterBi.NA
      #SQL { insert into Halkozko_txango_Bezero
      values (Bilbo', 2012-05-31, 12, nom)};
      }
      BufferedReader sarrera = new BufferedReader(new InputStreamReader(System.in));
      String DRIVER = "oracle.jdbc.driver.oracleDriver";
      String DBURL = "jdbc:oracle:thin:@000100.gi.en.es:1512:Ereala";
      String UID = "GRANITZALEA";
      String PWD = "PRAHITZA";
      try
      {
          /* Datu-basearekin konekzioa sortu eta konexio-testuinguru lehenetsia definitu */
          Driver controlador = (Driver) Class.forName(DRIVER).newInstance();
          DriverManager.registerDriver(controlador);
          DefaultContext test = new DefaultContext(DBURL, UID, PWD, false);
          DefaultContext.setDefaultContext(test);
          System.out.println("Konekzioa eginda");
          System.err.println("Errorea konekzioa: " + e.getMessage());
          System.exit(1);
      }
      catch (Exception e)
      {
          System.out.println("Errorea konekzioa: " + e.getMessage());
      }
      #SQL iterDat = {Select HTB.NA, telefono from Halkozko_txango_Bezero AS HTB Natural
      Join Bezero where HTB.Helmug = "Bilbo" AND TtxedoDate = "2012-05-31"
      AND IdentiLizsio = "5"};
      while (iterDat.next())
      {
          if (telefono != IterBi.telefono) == true ->
          nom = IterBi.NA, telefono from Bezero where NA := nom;
          #SQL iterBi = Select NA, telefono from Bezero
          #SQL {Delete from Halkozko_txango_Bezero where NA = nom};
      }
  }
```

#SQL {select transaction isolation level read committed;
 while (IterBi.next())
 row = IterBi.NA
 #SQL {Delete from Halkozko_txango_Bezero where NA = row};
 #SQL {Delete from Halkozko_txango_Bezero where NA = IterBi.NA};

Ariketak: 5 → 4 bidean

Ustez libre zegoen zerbait jadanik ez dago libre. Nola tratatu?

5.- Ostatuen erreserba

```
void pauso2( ) throws Exception {
    BufferedReader sarrera = new BufferedReader (new InputStreamReader (System.in));
    String na, izena, data, egunak, oizena;
    int kont, gkop, ekop = 0;
    Boolean libre?;
    try {
        while (true) {
            kont = kont + 1;
            ... // Datuak eskatu eta irakurri
            try {
                libre? = true;
                #sql { select gelakop into :gkop from Ostatua where Oizena = :izena };
                for (int i=0; i=egunak; i++) {
                    #sql { select count(*) into :ekop from Erreserba where Oizena = :izena and data = :data + :i };
                    libre? = (libre? and (gkop > ekop));
                }
                if (not libre?) {
                    System.out.println("Ostatua data horietan erreserbatuta dago");
                    System.out.println("Berriz saiatu");
                    if (kont < 3) continue;
                }
            }
            #sql { set transaction isolation level read committed };
            for (int i=0; i=egunak; i++) #sql { insert into Erreserba values (:na, :izena, :data + :i) };
            #sql { commit };
            break
        } catch (Exception e) { System.out.println("errorea: " + e.getMessage( )); }
    } catch (Exception e) {System.out.println("errorea: " + e.getMessage( )); }
```

4.- Proiektuak (murriztapen bat dauka definituta)

```
#sql {set transaction isolation level read committed};
for (i=0, i=2, i++) #sql {insert into matrikula values (:ident, :taula(i), null, null)};
#sql {commit};
break;
} catch (SQLException e) {
    #sql {rollback};
    System.err.print (" Aukeraketa prozesua berriz hasi beharko da ");
    if (e.getErrorCode( ) == "CONSTRAINT_VIOLATED") {
        System.out.println ("Aukeratutako proiekturen bat jadanik ez dago libre.");
        if ( lag = 3) break;
        else continue;
    }
}
...
...
```



```
        System.out.println ("Egindako aukeraketa guzlia gorde da ");
        break;
    } catch(e SQLException){
        #sql {rollback};
        System.err.print (" Aukeraketa prozesua berriz hasi beharko da ");
        if (e.getErrorCode( ) == "CONSTRAINT_VIOLATED") {
            System.out.println ("Aukeratutako proiekturen bat jadanik ez dago libre.");
            if ( lag = 4 ) break; /* Aukeraketa lau alditan bakarrik egin dezake*/
            else continue;
        }
    }
}
} // if
} // while
} catch(SQLException e){ System.err.print (" Errorea gertatu da "); }
}
```

```

import java.sql.*;

#sql iterator IterKlasea (String izena);           // iteratzalea SELECT galderentzat
class ProiekturenAukeraketa {
public static void main ( String args[ ] ) {
    IterKlasea iterBat, iterBi;
    String ident, proiek, sailBat;
    String[ ] taula = new String[3];
    int kont, lag;
    BufferedReader sarrera = new BufferedReader (new InputStreamReader (System.in));

    try {
        try { /* DBrekin konexioa sortu eta lehenetsitako konexo testuingurua definitu */
            Driver kontrolatzalea = (Driver) (Class.forName("oracle.jdbc.driver.OracleDriver").newInstance( ));
            DriverManager.registerDriver(kontrolatzalea);
            DefaultContext test = new DefaultContext("jdbc:oracle:thin:@jiplc0.si.ehu.es:1512:erreala", "erabil", "nerea", false);
            DefaultContext.setDefaultContext(test);
            System.out.println ("Konexioa eginda");
        } catch(Exception e) {
            System.err.print ("Errorea konexioan " + e.getMessage());
            exit(-1);
        }

        System.out.print ("Ikaslearen NA: "); ident = sarrera.readLine();
        /* Ikaslek proiekturen aukeraketa egina al du? */
        #sql iterBat = {select proiek as izena from matrikula where ikas = :ident};
        if (!iterBat.next()) { /* Ikaslek oraindik ez du proiekturek aukeratu, proiekturen zerrenda atera */
            iterBat.close();
            lag = 0;
            while (true) {
                try {
                    lag = lag + 1;      /* Sail guztien zerrenda */
                    #sql iterBat = {select izena from saila order by izena};
                    while (iterBat.next()) {
                        sailBat = iterBat.izena();
                        System.out.println ("SAILA: " + sailBat);
                        /* 5 ikaslek baino gutxiagok aukeratu dituzten proiekturen zerrenda */
                        #sql iterBi = {select proiek as izena
                                      from matrikula right join proiekta on proiek = titulua
                                      where bereSaila = :sailBat
                                      group by proiek
                                      having count (ikas) < 5};
                        kont = 0;
                        while (iterBi.next()) {
                            System.out.println (" " + iterBi.izena());
                            kont = kont + 1;
                        }
                        if (kont == 0) System.out.println ("Sail honek ez du proiektu librerik");
                        iterBi.close();
                    }
                    iterBat.close();
                    /* Ikasleari bere proiektauak eskatu, bitarteko taulan gorde, eta
                     gero insert guztiak bata bestearen ondoren egin*/
                    for (int i=0; i<2; i++) {
                        System.out.print ("Eskatutako proiektauren izena: ");
                        taula [i] = sarrera.readLine();
                    }
                }
                #sql {set transaction isolation level read committed};
                for (i=0, i<2, i++) #sql {insert into matrikula values (:ident, :taula(i), null, null)};
                #sql {commit};
            }
        }
    }
}

```

TRANSAKZIOAK

- ### 1. Hasierako baliak: X=1000; N=300; M=100.

(b) Eta ze balio beharko luke? ||go
(c) Ze kasuaren aurrean gaude:

- (b) Eta ze balio beharko luke? 
 (c) Ze kasuaren aurrean gaude:

T1 transakzioa: $A:=A+1$; write(A); read(B); read(A); B:=B/A; write(B); commit;	T2 transakzioa: read(A); A:=A*2; write(A); commit;
set transaction isolation level read uncommitted	T2 transakzioa

		U. Eguneratze galdutaren arazoa
ii.	Bchin-betineko aldatzearen arazoa	$A := A + 1$
iii.	Kalkulu okerraren arazoa	write(A)
iv.	Irakurketa errepikakizma	read(A)
v.	Ez da molako arazorik egin	$A := A * 2$
Isterako baliokak: X=1000; N=300; M=100.		
T1 transakzioa		
read(X)	set transaction isolation level read uncommitted	
X:=X+N		
write(X)		
T2 transakzioa		
		write(A)
B:=B/A		
write(B)		
commit		
(a) Ze gorputzen da osiltzen ahalmena?		commit

2. Hasierako baliok: X=1000; N=300; M=100

asierako baliok: $X=1000$; $N=300$; $M=100$

T1 transakzioa	T2 transakzioa
read(X)	set transaction isolation level read uncommitted
X:=X+N	
write(X)	
commit	
	read(X)
	X:=X+M
	write(X)
	commit

(c) Ze Kasten auf; ein gäde:

3. Hasierako balioak: X=1000; N=300; M=100

ierako balioka: X=1000; N=300; M=100

- (a) Bi transakzioak exekutatu osjean ze hallo adur-tildeku Γ \times Γ -tik \rightarrow Γ -ra

a) Bi transakzioak exekutatu ostean ze hain zuzendaketa erabili.

T1 transakzioa	T2 transakzioa
set transaction isolation level read committed	
(a) select id from erabilizale	
(b) select id from erabilizale	insert into erabilizale values (3, Axun, 23)
(c) select id from erabilizale	commit
commit	

6. ERABILITZAILE

id	izena	adina
1	Jon	20
2	Mander	25

T1 transakzioa	T2 transakzioa
set transaction isolation level read uncommitted	
(a) select id from erabiltzaile	insert into erabiltzaile values (3, Axun, 23)
(b) select id from erabiltzaile	commit
(c) select id from erabiltzaile	
commit	

Zer bueltatuko du SELECT bakoitzak:

- (a) select id from erabiltzaile \, 1, 2
- (b) select id from erabiltzaile \, 1, 2, 3
- (c) select id from erabiltzaile \, 2, 3

7. Ze gertatzen da honako exekutazio honetan:

T1 transakzioa	T2 transakzioa
lock x (B)	
read (B)	
B = B+10	
write (B)	
	lock s (A)
	read (A)
	lock s (B)
lock x (A)	
...	

Elkar blokeatuta da debo

lock x (B)

lock s (A)

lock s (B)

8. Honako sententziak exekutatu ostean...

```

DefaultContext ing = new DefaultContext(
    "jdbc:odbc:adbidea", "Ni", "auskalo", false);
DefaultContext.setDefaultContext(ing);
#sql {insert into Saila values (100, 'PAYROLL');
#sql {insert into Saila values (200, 'ACCOUNTING');
#sql {rollback};
testing.getConnection().setAutoCommit(true);
#sql {insert into Saila values (300, 'MARKETING');
#sql {rollback};
#sql {insert into Saila values (400, 'BRAND');
#sql {commit};
#sql {insert into Saila values (500, 'CLIENT ACCOUNT');
#sql {rollback};
#sql

```

- a) Zenbat tupla dago datu-basean? 3 tuple
- b) Zein dira tupla horiek?

300 Marketing

400

Brand

500

Client Account

49. Arikete

Select Helmuge, IntercDate

From Bidarie

Where Eiguren * EigurenPresia <= ALL (Select Eiguren * EigurenPresia

From Bidarie)

50. Arikete

Select Helmuge, IntercDate, Zenbaki

From Hezketao_txango_Bedero AS HBB

Group By Helmuge, IntercDate, Zenbaki

Having Count(*) >= ALL (Select Count(*)

51. Arikete

Select NA, Izena

From ~~Bidarie~~ Natural Join

(Bedero Natural Join Hotel_Bidarie_Bedero) AS HBB Group By HBB.Helmuge, HBB.IntercDate, HBB.Zenbaki

INNER JOIN Bidarie ON

HBB.Helmuge = B.Helmuge AND HBB.IntercDate = B.IntercDate

Where Eiguren * EigurenPresia <= ? AND B.IntercDate <= ?

Except

Select NA

From HBB ~~INNER JOIN~~ Bidarie

Where Eiguren * EigurenPresia <= 3000

52. Arikete

Select NA, Izena

From Bidarie AS G

Where Not Exists (Select *

From Bidarie AS B

B.NA <> G.NA)

Where Eiguren > 50 AND NotExists &

(Select *

From Bidarie AS B1

Where G.NA =

53. Arikete

Select NA, Izena

From Gideli AS G Natural Join Bidai AS B

Where B.Helbidea like 'Errone' AND B.IsteaDate like = '12-01-01'

Except

Select NA,

From Bidai

Where B.Helbidea like 'Errone' AND B.IsteaDate = '12-02-01'

54. Arikete

Select NA, Izena

From Gideli AS G Natural Join Bidai AS B

Where Helbidea like 'Errone'

Except

Select NA

From Bidai

Where Helbidea <> 'Errone'

55. Arikete

Select Helbidea, IsteaDate, Zerubetia

From Herribidea_txango AS Ht Left Join Herribidea_txango_Bezero AS HtB

ON Ht.Helbidea = HtB.Helbidea AND Ht.IsteaDate = HtB.IsteaDate AND

Ht.Zerubetia = HtB.Zerubetia

Where NA IS Null

56. Arillete

Select NA, Izene

From Gideri AS G

Where Hilmutze Lille 'Italiere' AND Exist (Select *
Group By From Bideie AS B)

Where 'Helmuge Lille 'Errour' AND G.NA = B.NA)

57. Arillete

Select NA, Izene

From Bezero AS B

Where Exist (Select *

From Hotel-Bideie-Bezero AS HBB

Where HBB.NA = HBB.H / AND HBB.Helmuge Lille 'Paris'
AND Not Exist (Select *

From Hotel AS H

H.Hinie \leftrightarrow 'Paris' AND H.HelId = HBB.HelId

Select Helmuge, InterDate, Izene

From Bideie AS B

INNER JOIN (Hotel-Helmuge AS HT ON B.Helmuge = HT.Helmuge AND

B.InteractionDate = HT.InteractionDate

Where Count(NA)

Select Helmuge, InterDate, Count(NA)

From Hotel-Helmuge AS HT Natural Join Hotel-Bezero
Where

Group By Helmuge, InterDate, Izene

Having Count(NA) \geq All / Select Count(NA)

= (Select max from

From Hotel-Helmuge Natural Join Hotel-Helmuge-Bezero
Where HT.Hel = HT1.Hel AND HT.Id \neq HT1.Id
Group

58. Arillete

53. Arikete

Select NA, Izene

From Gideri AS G Natural Join Bidcie AS B

where B.Helmege Like 'Erone' AND B.Isteerde Like '12-01-01'

Except

Select NA, Izene

From Gideri AS G1 Natural Join Bidcie AS B1

where B.Helmege Like 'Erone' AND B.Isteerde Like '12-02-01'

54. Arikete 51

Select NA, Izene

From Gideri AS G ~~Natural Join~~ Bidcie AS B

where ~~Kontakte~~ ~~(Select)~~

Helmege IN (~~From Bidcie~~
Select Helmege
From Bidcie

where Helmege Like 'Erone' AND G.NA = B.NA

55. Arikete

Select ~~Htak~~ Helmege, Isteerde, Beberin

From Herkales_Tango_Bebere ^{AS HTB} Right Join Herkales_Tango ^{AS HT} ON
HtB.Helmege = HT.Helmege AND HtB.Isteerde = HT.Isteerde AND HtB.Beberin = HT.

where NA IS NULL

49. Aril eta

Select Helmuga, IntercetData, Egunel * Egunel * Precision
 From Bidaia

Where

Egunel * Egunel * Precision <= ALL (Select Egunel * Egunel * Precision
 From Bidaia)

50. Aril eta

Select Helmuga, IntercetData, ^{Zurbelha}Count (NA)
 From Hotelak - Txango - Bezero

Where Count (NA) >= ALL (Select Count (NA))

From Hotelak - Txango - Bezero,

Group By Helmuga, IntercetData, Zurbelha

51. Aril eta

Select NA, Izenea

From Bezero ^{AS B}

Where Not Exists (Select *

> 3000

EXCET - NAT IN - #

< 3000

From Hotel - Bideia - Bezero AS HBB

Where B.NA = HBB.NA AND

Not Exists (Select *

From Bideia AS B1

Where B1.Helmuga = HBB.Helmuga AND

B1.IntercetData = HBB.IntercetData AND

Egunel * Egunel * Precision > 3.000))

52 Aril eta

Select NA, Izenea

From Gideri AS G

Where Not Exists (Select *

From Bidaia AS B

Where Egunel > 50 AND

Not Exists (Select *

From Gideri AS G1

Where B.NA = G.NA)

Riderak lotzen da AND ID

39. Ariketa

Select NA, Izene

From Bezero ASB

where Exist (Select*)

From Hotel_Bideia - Bezero AS HBB Natural Join Hotel AS H

where Izene.Hiztegia = Peris AND HBB.NA = B.NA AND

Not Exist (Select*)

From Hotelero_txango - Bezero AS HTB

where HTB.Helbidea = HBB.Helbidea AND

HTB.Isteeradate = HBB.Isteeradate AND B.NA = HBB.NA

47. Ariketa

Select HT.Helbidea, HT.Isteeradate, HT.Zerubetegia, Count(HT.NA)

From Hotelero_txango AS HT Left Join Hotelero_txango - Bezero AS HTB ON

HT.Helbidea = HTB.Helbidea AND HT.Isteeradate = HTB.Isteeradate AND

HT.Zerubetegia = HTB.Zerubetegia

Group By HT.Helbidea, HT.Isteeradate, HT.Zerubetegia

48. Ariketa

Select HT.Helbidea, HT.Isteeradate, HT.Zerubetegia, Count(HT.NA)

From ~~From~~ Hotelero_txango AS HT Left Join Hotelero_txango - Bezero AS

ON HT.Helbidea = HTB.Helbidea AND HT.Isteeradate = HTB.Isteeradate

AND HT.Zerubetegia = HTB.Zerubetegia

Where Count(HT.NA) ≤ 5

Group By HT.Helbidea, HT.Isteeradate, HT.Zerubetegia

45. Arilleta

Select *

From Bidaia^N Left JOIN Hauteko-Txango AS HTB ON

HTB.Helmuge = HTB.Helmuge AND HTB.IsteeraData = HTB.IsteeraData

Group By HTB.Helmuge, HTB.IsteeraData

46. Arilleta

From HTB

Select * HTB, HTB.NA, HTB.Isteera

From Hauteko-Txango AS HT left Join (Hauteko-Txango-Bedero^{HTB} Natural Join

Bedero AS B) ON

HTB.Helmuge = HTB.Helmuge AND HTB.IsteeraData = HTB.IsteeraData AND

HTB.Zerbeltic = HTB.Zerbeltic

Group By HTB.Helmuge, HTB.IsteeraData, HTB.Zerbeltic

38. Arilleta

Select NA, Isteera

From Bedero AS B

Where Not Exist (Select *

From Hotel-Bideia-Bedero AS HBB Natural Join Hotel AS H

Where H.Harie = Paris AND HBB.NA = B.NA

AND Not Exist (Select *

From Hauteko-Txango-Bedero AS HTB

Where HTB.Helmuge = HBB.Helmuge

AND HTB.IsteeraData = HBB.IsteeraData AND
HTB.NA = B.NA)

AND Exists (Select *

From Hotel-Bideia-Bedero AS HBB Natural Join Hotel AS H

Where H.Harie = 'Paris' AND B.NA = HBB.NA)

42. Ariketa

Select NA

From Hotel_Bideia - Bezero

Where Helmuga Like 'Funchal*

Group By NA, Helmuga, IrteeraDate

Having Count(*) >= (Select Count(HotelID)

From Hotel

Where Helmuga Like 'Funchal'

43. Ariketa

Select Hotel.ID, Count(*)

From Hotel_Bideia - Bezero AS HBB Natural JOIN Hotel

Group By HotelID

Having Count(*) >= ALL (Select Count(*)

From Hotel_Bideia - Bezero AS HBB Natural JOIN Hotel

Group By HotelID

Having Count(*)

44. Ariketa

Select NA, Izena, Sum(Precio)

From (HotelBideia - Bezero AS HTB Natural JOIN Bideia) Natural JOIN

Bideia AS B

Group By Helmuga, IrteeraDate

Having Sum(Precio) >= ALL (Select Sum(Precio))

From HotelBideia - Bezero Natural JOIN

Bideia }

Group By Helmuga, IrteeraDate

Having Sum(Precio)

36. Arikete

Select NA

From Bidai, Gidari

where Unique (Select NA

From Bidai

where Helmuge Lille Paris AND Gidari.NA ≠ Bidai.NA)

37. Arikete

Select NA

From Gidari

where Not Unique (Select NA

From Bidai

where Helmuge Like 'Paris' AND Gidari.NA = Bidai.NA)

40 Arikete

Select Helmuge, Irteeradeta, Zerbalkia

From Hartealbo-Txingo-Bezero

Group By Helmuge, Irteeradeta, Zerbalkia

Having Count(*) > 30

41. Arikete

Select Helmuge, Irteeradeta, Zerbalkia

From Hartealbo-Txingo Bidai

Group By Helmuge, Irteeradeta

Having Helmuge, Irteeradeta IN (Select Helmuge, Irteeradeta

From Hartealbo-Txingo
where

From Hartealbo-Txingo

Group By Helmuge, Irteeradeta

Having Count(Distinct txangotxeluze) = 1

32. Athlete

Select Helmuge, IntercDate

From Bidaiia

Where EguenekoPrezio > Any (Select EguenekoPrezio

From Bidaiia

Where Helmuge Like 'Ergome')

33. Athlete

Select Helmuge, IntercDate

From Bidaiia

Where Prezio * EguenekoPrezio >= ALL (Select Prezio * EguenekoPrezio
From Bidaiia)
Where

34. Athlete

Select Helmuge, IntercDate.

From Bidaiia AS B1

Where EguenekoPrezio <= ALL (Select Prezio

From Haurkalde_Berango AS HT

Where B1.Helmuge = HT.Helmuge

AND B1.IntercDate = HT.IntercDate)

35. Athlete

Select NA

From Bezero

Where NA <> ALL (Select NA,

From Hotel_Bidaiia_Bezero)

Where Helmuge Like 'Asilah' AND IntercDate ~~= 1/1/10101~~

26. Aritleta

Select izena

From Bezero Natural JOIN Hotel_Bideia_Bezero

Where

Where HotelID IN (Select HotelId

From Hotel

Where Hiria Like 'Durango')

27. Aritleta

Select NA

From Bideia Bezero HotelBideiaTango_Bezero

Where NA IN Not (Select NA

From HotelBideiaTango_Bezero)

29. Aritleta

Select Helmuga, InteresData

From Bideia

Where EgunKop, ^{Helmuga}InteresData IN (Select DatEgunKop, ^{Helmuga}InteresData

ALL ETA ~~ANY~~

31. Aritleta

Select Helmuga, InteresData

From Bideia

Where Egunelko Precio > ALL (Select Egunelko Precio

From Bideia

Where Helmuga Like 'Erromel')

16. Arikete

Select Count (Distinct NA)

From Bidic

where Helnuge Lille 'Paris'

17. Arikete

Select NA, Count(Distinct NA)

From Hibantuze

Group BY NA

18. Arikete

Select Helnuge, Count(Distinct B.NA)

From (Bideic AS B Natural JOIN Hotel_Bideic) Natural JOIN
(Hotel_Bideic Natural JOIN Hotel_Hire)

where Hibantuze.Hibantuze.Lille Italiere AND Hotel.Hire.Lille 'Grono'

Group By B.Helnuge

225.

Exists etc IN

25. Arikete

Select Distinct NA, Szena

From Gideri

where NA IN (Select NA

From Bideic

where Egnade.Predio * Egnade > 100

Is NULL, IN, Like etc Between

7. Arithmetic

Select Helmuge, Interedate
 From Bidic
 Where NA is NULL

Union, except and intersect

11. Arithmetic

Select ~~Iden~~. Den
 From Gideri
 Union
 Select ~~Bzzer~~. Den
 From Bezero

13. Arithmetic

Select Iden
 From Gideri
 Except
 Select Iden
 From Bezero

Function aggregately etc Group By

14. Arithmetic

Select AVG (Precice * Egunclo Precice)
 From Bidic
 Where Interedate Like 'Domestic'

Datu-baseak: SQL Ariketa

INNER JOIN eta Auto JOIN

3. Ariketa

Select Helvuga, IrteeraData

From Bidaria As B1 INNER JOIN Hautsak - Txango As HT ON

B1.Helvuga = HT.Helvuga AND B1.IrteeraData = HT.IrteeraData

where HT.Txango.Helvuga like 'toledo' AND B1.Helvuga like 'Madril'

AND B1.IrteeraHiria like 'Domestik'

5. Ariketa

Select Helvuga, IrteeraData, NA Izene

From (Bidaria As B1 INNER JOIN Gideri AS G ON B1.NA = G.NA)

INNER JOIN (Bezero As Be INNER JOIN Hotel_Bidaria - Bezero AS HBB ON

HBB.NA = Be.NA) ON B1.Helvuga = HBB.Helvuga AND B1.IrteeraData =
HBB.IrteeraData

~~txango~~

6. Ariketa

Select Izene NA

From (Hotel_Bidaria - Bezero As HBB INNER JOIN Hotel As H

ON H.HotelId = HBB.HotelId) INNER JOIN (Hotel_Bidaria - Bezero NATURAL JOIN
Hotel) ON H.GeluBopurua < Hotel.GeluBopurua

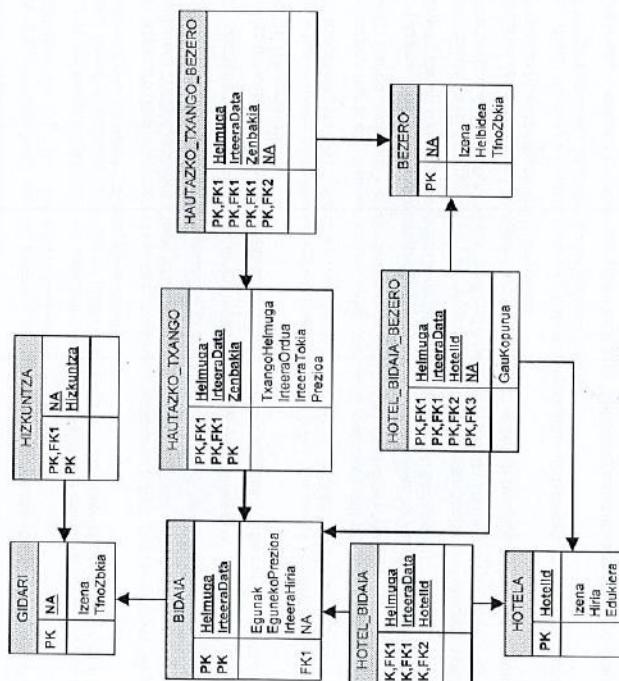
where HBB.IrteeraData like '2011-0503' AND HBB.NA like '15238432'
AND Hotel.Izene like 'Gabrielle'

Datu-baseak: SQL ariketak

Datu-baseak: SQL ariketak

Lanetik ereritutti dauen personentzat egun bat baino gehiago irauten duten bidaia antolatzentzutu ONDO-PASA
enpresak, datuak kideanteko dituen aplikazio informatibo bat garatu du.

- Bidaia bat lantzeordez, Benidorm-en) data desberdinetan egin daiteke. Bidaia bakoitzko identifikatzeko helmuga eta irteera-data erabilizten dira. Gainera, bidaiaen iraupena eginetan, eguneko prezioa, irteera-hiria, erabilgari eta bidaia hotelak em bidaian estututako gidiani bakarra gordetzen dira.
- Gidari bakoitzak bere NA-arekin identifikatzen da eta bere izena, telefono-zenbakia eta hitz egiten dituen hizkuntzak ezagutzen dira.
- Hotel bakoitzeko here identifikazio zenbakia bakarra, izena, hiria eta edukiera gordetzen dira.
- Bezeroen, NA-arekin identifikatzen, izena, helbidea eta telefono-zenbakia ezingozen dira.
- Bidaia datean parte hartzen duen bezero bakoitzeko zenbat gau pasatzen bezeroak bidaia horri esleitutako hoteltzat gorde nahi da. Bidaia egun bat baino gehiago irauten duten gertakasak bidaia berean hotel bat baino gehiago esleitu izana.
- Bidaia bakoitzak hautazko txango bat edo gehiago eduki ditza. Hautazko txango bakoitzaren ordenanzak (1., 2., 3., ...) helmuga, prezioa, irteera-ordua eta irteera-tokia gordetzen dira. Zein bezeroek egin duten hautazko txango bakoitzak ere gordetzen dira.
- Hotel bakoitzeko here identifikazio zenbakia bakarra, izena, hiria eta edukiera gordetzen dira.
- Bidaia datean parte hartzen duen bezero bakoitzeko zenbat gau pasatzen bezeroak bidaia horri esleitutako hoteltzat gorde nahi da. Bidaia egun bat baino gehiago irauten duten gertakasak bidaia berean hotel bat baino gehiago esleitu izana.
- Bidaia bakoitzak hautazko txango bat edo gehiago eduki ditza. Hautazko txango bakoitzaren ordenanzak (1., 2., 3., ...) helmuga, prezioa, irteera-ordua eta irteera-tokia gordetzen dira. Zein bezeroek egin duten hautazko txango bakoitzak ere gordetzen dira.



Gidari (NA, Izena, TnoZbka)
Bidaia (Helmuga, IrteeraData, Eguna, EgunekoPrezioa, IrteeraHiria, NA)
Hotela (HotelId, Izena, Hiria, Edukiera)
Bezero (NA, Izena, Helbidea, TnoZbka)
Hotel_Bidaria_Bezero (Helmuga, IrteeraData, HotelId, NA, GauKopurua)
Hautazko_Txango_Bezero (Helmuga, IrteeraData, Zentzia, IrteeraOrdua, IrteeraTokia, Prezioa)

INNER JOIN eta NATURAL JOIN

- Loritzazu italeraz hitz egiteko gaitasuna duten gidarien NA-a eta izena.
- Lortu Sevilla hiriko hotelak eskanitzentz dituzten bidaien kodea.
- Loritzazu honako baldintza betetzen duten bidaien kodea: Donostiatik atera eta Madril helmuga gisa eduki, eta Toledo hirira hautazko txangoak aukeratu dituzten bezeroen NA-a eta izena.
- Lortu bidaiaren batean hautazko txangoak aukeratu dituzten bezeroen NA-a eta izena.
- Loritzazu honako baldintza hau betetzen duten bidaien identifikazioa, eta bezeroen NA-a eta izena: bezeroaren izenak eta egindako bidaiaren gidariaren izenak berdinak izan behar dute (adibidez, Mikel, Mikel).

AUTOJOIN

6. Lortu 2011-05-03 datan Errromara egindako bidaian NA=15238432 daukan bezeroak Gabriella izeneko hotelean (garo zituen gauko baino gau gehiago hotel berean (bidaiar batean edo besteren bat) izenaren, eta ez nahitaez Gabriella hotelean) egoncako bezeroen izena eta helbidea.

IS NULL, IN, LIKE eta BETWEEN eragileak

- Loritzazu gitarririk gabeko bidaien kodeak.
 - Hotelaren egoitzak Miarritzen edo Baiionan duten bidaiai kodea lortu hotelaren izenarekin batera.
 - Loritzazu, errekapenik gabe, Zarautzen bizi eta bidaiaren bat egin duten bezeroen izena.
 - Loritzazu 2010. urtean zehar Donostiatik Madrilera egindako bidaia guztien irteera-data.
- UNION, EXCEPT eta INTERSECT**
- Lortu gidarien era bezeroen izenak.
 - Lortu izen bereko bezeroen eta gidarien izenak.
 - Lortu izen bereko bezeroen eta gidarien izenak.
- Funtzio agregatuak (COUNT, SUM, MAX...) eta GROUP BY**
- Loritzazu Donostiatik ateratzen diran bidaiai batez besteko prezioa.
 - Loritzazu 2011ko apirilean Parisera egindako bidaien kopurua.
 - Loritzazu Parisera bidaiaren bat egin duten gidiari desberdinen kopurua.
 - Gidari bakoitzeko kalikula ezzazu zenbat hizkuntza manperatzen dituen.
 - Helmuga bakoitzeko kalikula zenbat gidiari dagoeen, italierez hitz egiteko gaitasuna izanik, helmuga horretara biktariaren bat gidiatuko duena, baldin eta bidaiar horretan Errromako hotelen batean osoa hautzeko aukera badago.
 - Lortu hautazko txango garestiendari dagokion zenbakia.

[NOT] EXISTS eta [NOT] IN

- Lortu ezazu bidaiaren batean ere parte hartu ez duten gidarien NA-a.
- Loritzazu honako baldintza hau betetzen duten gidarien NA-a eta izena: euskaraz hitz egiteko gaitasuna izanik, Lekumberriko hotelen batean osoa hautzeko aukera baten gidaria izatea.
- Loritzazu ingelesez hitz egiten ez duten gidarien NA-a eta izena.
- Loritzazu Madrileko Ritz izeneko hotelean egoitzaz hartzeko aukera baten gidaria izatea.
- Loritzazu Madrileko Ritz izeneko hotelean egoitzaz hartzeko aukera baten gidaria izatea.

Datu-baseak: SQL ariketak

Datu-baseak: SQL ariketak

- ALL eta ANY**
25. Lor izazu egunean 100 euro baino gehiago balio duen bidaiaren batan gidian direnen NA-a eta izara.
 26. Lor ezazu honako baldintza hau betetzen duten bezeroen izena: bezeroak hotelaren egotiza Duran- gen duen bidaiaren batean parte hartu du.
 27. Lor ezazu hautazko txangorrik hartu ez dituzten bezeroen NA-a.
 28. Lor izazu honako baldintza hau betetzen duten Matrigo hotelen identifikazioa eta izena: hotelaren edukiera eta Macilgo "Florida Norte" hotelaren edukiera berdinak dira.
 29. Lor ezazu honako baldintza hau betetzen duten bidaien kodea: 2011ko urtarrilaren 7an Irunbelek ateratako bidaiaren helmuga bera eta egun kopuru berdina izatea.
 30. Hautazko txangorrik hartu ez duten bezeroen kopurua lor ezazu.

ALL eta ANY

31. Lor ezazu Erronkara egindako bidaia guztien eguneako prezioa gainditzen duten bidaien kodeak.
32. Lor ezazu Erronkara egindako bidaaren baten eguneko prezioa gainditzen duten bidaien kodea.
33. Lor ezazu bidaiaren jarestellenaren kodea.
34. Lor ezazu honako baldintza hau betetzen duten bidaien kodea: bidaiaren hautazko txango guztien prezioa bidaiaren eguneako prezioaren berdina edo handiagoa da.
35. Lor ezazu honako baldintza hau betetzen duten bezeroen NA-a: bezeroak ez zuen parte hartu 2011ko urtarrilaren 1ean Asiah helmuga gisa zeukan bidaien.

- [NOT] UNIQUE**
36. Lor ezazu Erronkara egindako bidaia guztien eguneako prezioa gainditzen duten bidaien kodeak.
 37. Lor ezazu bidaiaren helmuga Paris izanik bidaia bi edo gehiago jidatu dituzten gidianen NA-a.
- [NOT] EXISTS kabiakutak**
38. Lor izazu honako baldintza betetzen duten bezeroen NA-a eta izena: bezeroak hautazko txangoak hartu citu hotelaren egoitzak Parisen duden bere bidaia guztietan, zeinetaen hotelaren egoitza Parisen.
 39. Lor izazu honako baldintza hau betetzen duten bezeroen NA-a eta izena: bezeroak ez du inolako hautazko txangorrik hartu hotelaren egoitzak Parisen duden bere bidaia guztietan (gerta like, noski, bezero horrek hautazko txangoak Hartula Izatea hotelaren egoitza Parisen duden beste bidaia bidaaren batean. Baldintza bestea era batera esanda: bezeroak ez du inolako hautazko txangorrik hartu bere bidaaren batean, zeinetaen hotelaren egoitza Parisen baitago).

HAVING

40. Lor ezazu 30 bezero baino gehiago dauzkaten hautazko txangoen kodea.
41. Lor ezazu honako baldintza hau betetzen duten bidaien kodea: bidaaren hautazko txango guztiek helmuga berbera caukate.
42. Lor ezazu honako baldintza betetzen duten bezeroen NA-a: bezeroak Funchalera egin dituen bildak kontuan hartuta hiri horretako hotel guztietan hartu du egoitzak.
43. Bezeroek egun dituzten bidaia guztiek kontuan hartuta, lortu hotel hautatuenaren izena eta esan zentzat aldiiz izan den hautatua.
44. Lor izazu honako baldintza betetzen duten bidaien gildarren NA-a eta izena, eta bildak kontuan hartuta txangorrik hartu ez dituzten prezioaren batura: bidaia garestiena izatea, hautazko txangoen prezioen baturak erreperezatu.

OUTER JOIN eta UNION JOIN

45. Lor izazu bidaia guztien datu guztiek (BIDALIA taulan jasotzen direnak), eta, bidaiek hautazko txangoak baditu, lortu txangoen datu guztiek ere (txangorrik ez badu, NULL atera beharko litzateke txangoen lotutako datuetan).
46. Lor izazu hautazko txangoen datu guztiek, eta, txango horiek bezeroek aurkaratuak izan badira, lor izazu, halaber, bezeroak NA-a eta izena (ez badira aukeratuak izan, bezeroen bi zutabe horietan NULL atera beharko litzateke).
47. Lor izazu hautazko txango guztian identifikazioa eta zenbat bezeroek hautatu duen txango bakoitzak.
48. Best bezero baino gutxiagoko hautazko txango bakoitzeko, lortu, txangoparen identifikazioaz gain, txango hon hautatu duten bezeroen kopurua. Txangoa bezerorik gabe geratu bada ere, emaitzan alerako da, kopurua 0 dela.

BESTE BATZUK

49. Lor ezazu bidaia markenaren kodea.
50. Hautazko txangoak eskantzen dituen bidaia bakoitzeko lor ezazu bidaia horretan bezaro gehien daizkin hautazko txangoaren kodea.
51. Lor izazu honako baldintza hau betetzen duten bezeroen NA-a eta izena: beti hautatu izan dituzte 3000 euro baino gehiago bakoitzak.
52. Lor izazu honako baldintza hau betetzen duten gildariaren NA-a eta izena: gildari horrek iraupen luzeago (30 egun baino gehiago) bidaia guztiek gida du (gerta likek bidaia horretaz gain iraupen luzeagoa ez den bidaiaren bat ere jidatu izana).
53. Lor izazu 2012ko urtarrilaren 1ean Erronkara egindako bidaia jidatu zuen gildariaren NA-a eta izena, baitan eta 2012ko otsailaren 1ean Erronkara egindako bidaia jidatu ez bazuen.
54. Lor izazu Erronkara egindako bidaiai baino jidatu ez dituzten gidianen NA-a eta izena
55. Lor ezazu bezeroenik gabe geratu dirsen hautazko txangoen identifikazioa.
56. Lor izazu Italiara hitz egiteko gaitasuna izanik Erronkara egindako bidaiai bat jidatu ez duten gidianen NA-a eta izena.
57. Lor izazu honako baldintza betetzen duten bezeroen NA-a eta izena: Parisera bidaiai batean parte hartzea, eta bidaia honi lehako bakoitzean hiri horretako hotelietan baino ez egotea.
58. Hautazko txangoak dauzkan bidaia bakoitzeko lortu bidaiai identifikazioa, bidaia horretan hautazko txangoen guztiek jidatu dituen bezeroaren NA-a eta zerbait diren hautazko txango horiek.
59. Lor izazu, ingelesez hitz egiteko gaitasuna izanik, ez Kanadarako eta ez Australiarako, duden gidianen NA-a eta izena (not: ez Kanadarako eta ez Australiarako),
60. Hiru bidaia baino gutxiago jidatu dituen gildari bakoitzeko, lortu gildariaren NA-a eta jidatu dituen bildaien kopurua. Gildarik bildatik jidatu ez badu ere, emaitzan alerako da, kopurua 0 dela.

61. Arilkote

~~Select NA AS Navigating_History
From Gidcar AS G ~~left join~~ History AS H
Left Join Pay
Group By History
Select Having Count(NA) <= ALL (Select Count(NA))~~

Select
From 6
Where

Select Ad FROM
FROM Bezero INNER JOIN Hotel_Bedrijf_Bedew
WHERE HBB.HotelNaam = 'Paris' AND NOT EXISTS (Select *
FROM

63. Arbeit

Not Exist
Bids

Bidirectional

NA = NA AND GROWNE

Exist

Bidace

Ercole

64. Arbeit

Select Helix, Interpolate, NA
From Bidec
Where Four Is *T 1.0

$\gamma = \text{AVG}(\text{Eguide}^* \text{tgguide.Precio})$

~~$\#4 \text{ AVG}(\text{Select Avg}(\text{Eguide}^* \text{tgguide.Precio})$
From Bidone)~~

64. Architec

Select Helvete, Interstate, BNA, Bore

From Bideie Natural Join Gideri

Where E.Equale * Eguale.Predec >= AKG (Eguale * Eguale.Predec)

54. Arillete

Select NA, Idem

From Gideria AS G Natural Join Bideie AS B

Where B.Helmuge = 'Erromic' AND G.NA = B.NA

Except

Select, NA

From Bideie

Where B.Helmuge <> 'Erromic' AND G.NA ≠ B.NA

55. Arillete

Select Helmuge, IrteerDate, ~~zurbaln-~~

From Hurbaln-Txango-Bederu AS HTB

Externe

Group By Helmuge, IrteerDate, ~~zurbaln-~~

Having Count(NA) >= ALL (Select Count(NA))

From Hurbaln-Txango-Bederu AS HTB1

Where HTB.Helmuge = HTB1.Helmuge AND
HTB.IrteerDate = HTB1.IrteerDate

Group By Helmuge, IrteerDate, ~~zurbaln-~~)

60. Arillete

Select NA, Count(B.Helmuge)

From Gideria AS G left Join Bideie AS B ON G.NA = B.NA

Where Count(Helmuge) < 3

51. Aritlect

Select NA, Izene

From (Bideie AS B Natural Join Hotel-Bideie - Bezero AS HBB) Natural Join
Bezero AS Be;

Except ~~Not Exists~~

where Egunole * Egunko Precision > 3000

Except

Select NA

From Bideie Natural Join Hotel-Bideie - Bezero

where Bezero.Egunole * Egunko Precision ≤ 3000

52. Aritlect

Select NA, Izene

From Gideric AS G

where Not Exists (Select *)

From Bideie AS B

where B.Egunole > 50 AND B.NA < G.NA)

Edo

Select NA, Izene

From Gideric AS G

where Not Exists (Select *)

From Bideie AS B

where B.Egunole > 50 AND

Not Exists (Select *)

From Gideric AS G1 Natural Join Bideie AS B1

where G1.NA = B.NA AND B1.Helminge = B.Helv
AND B1.InteractionDate = B.InteractionDate))

56. Arikete

Select NA, Izene

From Giederi ASG Natural Join Hialunza AS H

Where H.Hialunza.Lille 'Izene'

AND

Not Exists (Select →)

Hialunza Not Exists (Select Hialunza)

From Bidic ASB

Where B.Helunge.Lille 'Birone' AND G.NA = B.NA

Select NA, Izene

From Bezdrok ASB

Where HBB.Bezdrok.Lille 'Paris'

AND Not Exists (Select →)

From Hotel ASB

Where HBB.Hotel.Lille = Paris AND

B.NA = HBB.NA AND

Not Exists (Select →)

From Hotel

Where HBB.HotelId = Alt. Hotel Id AND

57. Arikete

Select Helunge, InsertDate, Zerbikie, NA, Count(NA)

From Hotello_Bezdrok_Bederg AS O

Group By Helunge, InsertDate, Zerbikie

Helunge Count(NA) >= Alt. (Count(NA))

From Hotello_Bezdrok_Bederg AS C
other O, H = 1, 4
Group By Helunge, InsertDate, Zerbikie

59. Anket

Select NA, Tren

From Gideri AS G Nested Join Hikayet AS H

where H.Hikayet Like 'İngiliz' AND

Not Exists (Select *)

From Bideci AS B

where B.Hikayet Like 'Australia' AND

B.B.NA = G.G.NA)

AND NOT EXISTS select *

From Bideci AS B

where B.Hikayet < > 'Karde' AND

B.NA = G.NA)

60. Anket

Select NA, Count(Helmege)

From Bideci AS B

Group By NA

Having < > (Count(Helmege))

From Bideci AS B

where B.NA = G.NA

Group By NA)

Select NA, Count(Helmege)

From Bideci left Join Gideri AS G

where Count(Helmege) < 3