

Zápočtový test (nejspíš z r. 2007)

1. Vypište názvy všech jídel stojících 100Kč.

```
SELECT nazev FROM JIDELNICEK WHERE cena=100;
```

2. Vypište různá jména všech hostů.

```
SELECT DISTINCT jmeno FROM host;
```

3. Zjistěte průměrnou cenu všech jídel na jídelníčku.

```
SELECT AVG(cena) FROM jidelnicek;
```

4. Zjistěte počet jídel, jejichž cena je mezi 100 a 150Kč.

```
SELECT COUNT(*) FROM jidelnicek WHERE cena BETWEEN 100 AND 150;
```

5. Zjistěte počet různých jmen všech hostů.

```
SELECT COUNT(DISTINCT jmeno) FROM host;
```

6. Zjistěte počet hostů, u kterých není vyplněna položka DATUM_NAROZENI.

```
SELECT COUNT(*) FROM host WHERE datum_narozeni IS NULL;
```

7. Zjistěte počet hostů, jejichž jméno začíná na L nebo v jejichž příjmení se nevyskytuje písmeno A.

```
SELECT COUNT(*) FROM host WHERE prijmeni LIKE 'L%' OR prijmeni NOT LIKE '%A%'
```

8. Zjistěte názvy nejdražšího a nejlevnějšího pokrmu.

```
SELECT MAX(cena), MIN(cena) FROM jidelnicek;
```

9. Pro každé číslo účtu vypište seznam objednaných pokrmů.

```
SELECT u.cislo_uctu, j.nazev FROM ucty u, polozka p, jidelnicek j WHERE u.cislo_uctu = p.cislo_uctu AND p.cislo_pokrmu = j.cislo_pokrmu;
```

10. Vypište seznam hostů s počtem jejich návštěv. Seznam seřadte sestupně podle počtu návštěv.

```
SELECT h.jmeno, h.prijmeni, COUNT(u.cislo_uctu) FROM host h, ucty u WHERE h.cislo_hosta = u.cislo_hosta GROUP BY h.cislo_hosta ORDER BY COUNT(u.cislo_uctu) DESC;
```

11. Zjistěte počet hostů, kteří nemají žádnou návštěvu restaurace.

```
SELECT COUNT(*) FROM host WHERE cislo_hosta NOT IN (SELECT cislo_hosta FROM ucty);
```

12. Vypište jména a příjmení hostů, kteří mají více jak 10 návštěv restaurace.

```
SELECT h.jmeno, h.prijmeni FROM host h, ucty u WHERE h.cislo_hosta = u.cislo_hosta GROUP BY h.cislo_hosta HAVING COUNT(u.cislo_hosta) > 10;
```

13. Přidejte nového hosta.

```
INSERT INTO host VALUES(100, 'petr', 'svetr', 1982);
```

14. Změňte novému hostu datum narození.

```
UPDATE host SET datum_narozeni = 1912 WHERE cislo_hosta = 100;
```

15. Zrušte všechny pokrmy levnější jak 30Kč.

```
DELETE FROM jidelnicek WHERE cena < 30;
```

16. Zdražte všechny pokrmy v jídelníčku o 10%.

```
UPDATE jidelnicek SET cena = cena * 1.1;
```

17. Vytvořte tabulku HOST se sloupci CISLO_HOSTA, JMENO, PRIJMENI a DATUM_NAROZENI. Číslo hosta je primární klíč, jméno a příjmení jsou povinné.

```
CREATE TABLE host (  
  cislo_hosta NUMBER(5) PRIMARY KEY,  
  jmeno varchar2(20) NOT NULL,  
  prijmeni varchar2(20) NOT NULL,  
  datum_narozeni DATE);
```

18. Vytvořte tabulku UCET. Primárním klíčem je CISLO_UCTU. Zajistěte referenční integritu, tj. k účtu musí existovat host.

```
CREATE TABLE ucet (  
  cislo_uctu NUMBER(5) PRIMARY KEY,  
  cislo_hosta NUMBER(5),  
  constraints p_ch FOREIGN KEY cislo_hosta  
  REFERENCES host (cislo_hosta)  
);
```

19. Napište, jakým způsobem lze zajistit, aby při rušení hosta byly automaticky zrušeny všechny jeho účty.

```
...  
constraints f_ch FOREIGN KEY cislo_hosta  
REFERENCES host (cislo_hosta)  
ON DELETE cascade  
...
```

20. Zajistěte, aby nebylo možné zadat do jídelníčku pokrm, jehož cena není mezi 10 a 1000 Kč.

```
CREATE TABLE jidelnicek (  
  ....  
  CENA NUMBER(4) CHECK (CENA >= 10 AND CENA <= 1000)  
);
```