**CHEAT SHEET**

**BEGINNER SYNTAX**

| **Syntax** | **Popis** |
| --- | --- |
| SELECT | Vyber stĺpce |
| FROM | Z ktorých tabuliek čítať dáta |
| WHERE | Filtrovanie dát |
| GROUP BY | Skupinové operácie |
| HAVING | Filtrovanie agregovaných dát |
| ORDER BY | Zoradenie dát |

**BEGINNER SYNTAX**

**1. SELECT**

SQL kód

SELECT stlpec1, stlpec2 FROM tabulka;

* **Vysvetlenie:** Vyberie a zobrazí hodnoty z dvoch stĺpcov stlpec1 a stlpec2 v tabuľke tabulka.

**Príklad:**

Tabuľka **produkty**:

| **ID** | **názov** | **cena** |
| --- | --- | --- |
| 1 | Notebook | 1000 |
| 2 | Telefón | 500 |
| 3 | Tablet | 700 |

Dotaz:

SQL kód

SELECT názov, cena FROM produkty;

**Výsledok:**

| **názov** | **cena** |
| --- | --- |
| Notebook | 1000 |
| Telefón | 500 |
| Tablet | 700 |

**Prečo je to tak:**

* Vyberá iba stĺpce názov a cena.
* Stĺpec ID nie je zahrnutý, pretože nie je uvedený v SELECT.

**2. FROM**

SQL kód

SELECT \* FROM tabulka;

* **Vysvetlenie:** Zobrazí všetky stĺpce a všetky riadky tabuľky.

**Príklad:**

Tabuľka **zamestnanci**:

| **ID** | **meno** | **pozícia** | **plat** |
| --- | --- | --- | --- |
| 1 | Jozef | Manažér | 2000 |
| 2 | Anna | Asistent | 1200 |

Dotaz:

SQL kód

SELECT \* FROM zamestnanci;

**Výsledok:**

| **ID** | **meno** | **pozícia** | **plat** |
| --- | --- | --- | --- |
| 1 | Jozef | Manažér | 2000 |
| 2 | Anna | Asistent | 1200 |

**Prečo je to tak:**

* Používa \*, čo znamená výber **všetkých stĺpcov** tabuľky.
* Zobrazia sa všetky údaje z tabuľky **zamestnanci**.

**3. WHERE**

SQL kód

SELECT \* FROM tabulka WHERE stlpec1 = 'hodnota';

* **Vysvetlenie:** Zobrazí iba riadky, kde hodnota v stĺpci stlpec1 je rovná 'hodnota'.

**Príklad:**

Tabuľka **produkty**:

| **ID** | **názov** | **cena** |
| --- | --- | --- |
| 1 | Notebook | 1000 |
| 2 | Telefón | 500 |
| 3 | Tablet | 700 |

Dotaz:

SQL kód

SELECT \* FROM produkty WHERE cena = 500;

**Výsledok:**

| **ID** | **názov** | **cena** |
| --- | --- | --- |
| 2 | Telefón | 500 |

**Prečo je to tak:**

* Filtrovanie podmienkou cena = 500.
* Zobrazený je iba riadok, kde má stĺpec cena hodnotu **500**.

**4. GROUP BY**

SQL kód

SELECT stlpec1, COUNT(\*) FROM tabulka GROUP BY stlpec1;

* **Vysvetlenie:** Skupinuje riadky podľa hodnôt v stlpec1 a počíta počet riadkov v každej skupine.

**Príklad:**

Tabuľka **objednavky**:

| **ID** | **zákazník** | **produkt** |
| --- | --- | --- |
| 1 | Jozef | Notebook |
| 2 | Anna | Telefón |
| 3 | Jozef | Tablet |

Dotaz:

SQL kód

SELECT zákazník, COUNT(\*) FROM objednavky GROUP BY zákazník;

**Výsledok:**

| **zákazník** | **COUNT(\*)** |
| --- | --- |
| Jozef | 2 |
| Anna | 1 |

**Prečo je to tak:**

* GROUP BY zákazník vytvorí skupiny podľa hodnoty v stĺpci zákazník.
* COUNT(\*) spočíta počet riadkov v každej skupine.

**5. HAVING**

SQL kód

SELECT stlpec1, AVG(stlpec2)

FROM tabulka

GROUP BY stlpec1

HAVING AVG(stlpec2) > 100;

* **Vysvetlenie:** Zobrazí skupiny, kde priemerná hodnota v stlpec2 je väčšia ako 100.

**Príklad:**

Tabuľka **predaje**:

| **ID** | **predajca** | **suma** |
| --- | --- | --- |
| 1 | Jozef | 150 |
| 2 | Anna | 80 |
| 3 | Jozef | 200 |

Dotaz:

SQL kód

SELECT predajca, AVG(suma) FROM predaje GROUP BY predajca HAVING AVG(suma) > 100;

**Výsledok:**

| **predajca** | **AVG(suma)** |
| --- | --- |
| Jozef | 175 |

**Prečo je to tak:**

* GROUP BY predajca skupinuje riadky podľa predajca.
* HAVING AVG(suma) > 100 filtruje iba skupiny, kde priemerná hodnota v suma je nad 100.

**6. ORDER BY**

SQL kód

SELECT \* FROM tabulka ORDER BY stlpec1 ASC;

* **Vysvetlenie:** Zoradí výsledky vzostupne podľa hodnôt v stĺpci stlpec1.

**Príklad:**

Tabuľka **produkty**:

| **ID** | **názov** | **cena** |
| --- | --- | --- |
| 1 | Notebook | 1000 |
| 2 | Telefón | 500 |
| 3 | Tablet | 700 |

Dotaz:

SQL kód

SELECT \* FROM produkty ORDER BY cena ASC;

**Výsledok:**

| **ID** | **názov** | **cena** |
| --- | --- | --- |
| 2 | Telefón | 500 |
| 3 | Tablet | 700 |
| 1 | Notebook | 1000 |

**Prečo je to tak:**

* ORDER BY cena ASC zoradí tabuľku podľa stĺpca cena vzostupne (od najnižšej po najvyššiu hodnotu).