Základy počítačové grafiky, Databáze SQL, start PC

Vysvětlete pojem rastrové grafiky a princip jejího definování, uveďté její výhody a nevýhody, možnosti

Získání, zpracování a příklady použití bitmapové/rastrové grafiky v každodenní praxi, uvedte přiklad souborových formátů digitální grafiky,\_ jcjich rozdíly, výhody /nevýhody a užití

Co to je SQL, popište strukturu SQL databáze (tabulky, relace), Ukažte jaké SQL příkazy obsahují jednotlivé podkategorie DDL, DML, DCL. Vysvětlete syntaxi základních přikazů (CREATE TABLE, INSERT, UPDATE, SELECT, COMMIT)

popište kontrolu technického vybavení počítače (POST) při zapnutí počítače

Rastrová grafika (také nazývaná bitmapová grafika) je způsob zobrazení obrazů pomocí mřížky malých barevných bodů, tzv. pixelů. Každý pixel má přiřazenu určitou barvu a společně tvoří obraz. Každý pixel má pevnou pozici a nemění svou velikost při zvětšení nebo zmenšení obrazu, což vede k tomu, že při zvětšení obrázku může dojít ke ztrátě kvality (pixelizaci).

### Princip definování rastrové grafiky

Rastrová grafika je definována pomocí mřížky pixelů, kde každý pixel je uložen s určitou barvou. Tento typ grafiky vyžaduje, aby bylo uloženo mnoho detailních informací o každém jednotlivém pixelu, což může vést k velkým souborům, zejména při vysokém rozlišení.

### Výhody a nevýhody rastrové grafiky

**Výhody:**

* Detailní zobrazení jemných přechodů a složitých barevných struktur, což je výhodné pro fotografie a realistické obrazy.
* Přesné zobrazení světelných a barevných přechodů.
* Možnost zpracování efektů, jako je stínování, rozostření nebo jiné fotorealistické úpravy.

**Nevýhody:**

* Při zvětšení nebo zmenšení obrázku dochází k pixelizaci a ztrátě kvality.
* Větší velikost souborů, zvláště u obrázků s vysokým rozlišením.
* Omezená možnost zvětšování, což ztěžuje práci s logy nebo ilustracemi, které by měly být použity v různých velikostech.

### Možnosti získání a zpracování rastrové grafiky

* **Získání:** Fotografováním digitálním fotoaparátem, skenováním papírových materiálů, tvorbou v grafických programech jako je Adobe Photoshop nebo GIMP.
* **Zpracování:** Editací a úpravou v programech pro úpravu fotografií a rastrové grafiky (např. Adobe Photoshop, GIMP, Corel PaintShop Pro), kde lze upravovat jas, kontrast, barvy, a přidávat různé efekty.

### Příklady použití rastrové grafiky

* Fotografie (profesionální i amatérské).
* Grafické prvky webových stránek.
* Realistické ilustrace, jako jsou textury a obrazy na pozadí.
* Obaly knih, plakáty, reklamní bannery.

### Příklady souborových formátů rastrové grafiky

1. **JPEG (JPG):**
   1. **Výhody:** Malá velikost souboru díky kompresi, vhodný pro fotografie a webové stránky.
   2. **Nevýhody:** Ztrátová komprese – při každém uložení ztrácí část kvality, méně vhodný pro grafiku s ostrými hranami.
2. **PNG:**
   1. **Výhody:** Bezztrátová komprese, podpora průhlednosti, vhodné pro grafiku s ostrými hranami a ilustrace.
   2. **Nevýhody:** Větší velikost souboru než JPEG, ne všechny starší prohlížeče podporují průhlednost.
3. **GIF:**
   1. **Výhody:** Podpora animací, malá velikost souboru pro jednoduché obrázky, podpora průhlednosti.
   2. **Nevýhody:** Omezená paleta 256 barev, nevhodný pro fotografie a složité obrázky.

**2**

SQL (Structured Query Language) je jazyk používaný pro správu a manipulaci s daty v relačních databázích. SQL umožňuje uživatelům vytvářet databáze a tabulky, vkládat, aktualizovat, mazat a vybírat data a řídit přístup k databázi.

### Struktura SQL databáze

SQL databáze je tvořena hlavně těmito základními prvky:

1. **Tabulky** - Základní struktura, která uchovává data v řádcích (záznamech) a sloupcích (atributech). Každá tabulka je určena svým názvem a obsahuje sloupce s různými datovými typy (např. čísla, text, data).
2. **Relace (vazby)** - Určují vztahy mezi tabulkami pomocí klíčů:
   1. **Primární klíč (Primary Key)** – Unikátní identifikátor pro každý záznam v tabulce. Každá tabulka může mít pouze jeden primární klíč.
   2. **Cizí klíč (Foreign Key)** – Odkazuje na primární klíč jiné tabulky a vytváří tak relaci mezi dvěma tabulkami, což umožňuje propojení dat mezi nimi.

### SQL příkazy podle kategorií

SQL příkazy se dělí do tří hlavních kategorií:

1. **DDL (Data Definition Language)** – Příkazy pro definici struktury databáze (např. vytváření a mazání tabulek).
   1. **CREATE** – Vytváří databázi nebo tabulku.
   2. **ALTER** – Mění strukturu existující tabulky.
   3. **DROP** – Smaže tabulku nebo databázi.
2. **DML (Data Manipulation Language)** – Příkazy pro manipulaci s daty v tabulkách.
   1. **INSERT** – Vloží nový záznam do tabulky.
   2. **UPDATE** – Aktualizuje existující záznam v tabulce.
   3. **DELETE** – Smaže záznam z tabulky.
   4. **SELECT** – Vybere data z jedné nebo více tabulek.
3. **DCL (Data Control Language)** – Příkazy pro řízení přístupu a práv.
   1. **GRANT** – Uděluje uživateli oprávnění k určitým operacím.
   2. **REVOKE** – Odebere oprávnění od uživatele.
   3. **COMMIT** – Potvrdí změny provedené na datech.
   4. **ROLLBACK** – Vrátí změny zpět, pokud nebyly potvrzeny.

### Syntaxe základních SQL příkazů

#### 1. **CREATE TABLE**

Příkaz pro vytvoření nové tabulky v databázi.

CREATE TABLE Zamestnanci (

ID INT PRIMARY KEY,

Jmeno VARCHAR(50),

Prijmeni VARCHAR(50),

Pozice VARCHAR(50),

Plat DECIMAL(10, 2)

);

Tento příkaz vytvoří tabulku Zamestnanci se sloupci ID, Jmeno, Prijmeni, Pozice a Plat, kde ID je primární klíč.

#### 2. **INSERT**

Příkaz pro vložení nového záznamu do tabulky.

INSERT INTO Zamestnanci (ID, Jmeno, Prijmeni, Pozice, Plat)

VALUES (1, 'Jan', 'Novák', 'Programátor', 45000.00);

Tento příkaz vloží nový záznam do tabulky Zamestnanci s hodnotami ID = 1, Jmeno = 'Jan', Prijmeni = 'Novák', Pozice = 'Programátor', Plat = 45000.00.

#### 3. **UPDATE**

Příkaz pro aktualizaci existujícího záznamu v tabulce.

SELECT Jmeno, Prijmeni, Pozice

FROM Zamestnanci

WHERE Plat > 40000;

Tento příkaz vybere jméno, příjmení a pozici zaměstnanců, jejichž plat je vyšší než 40000.

#### 5. **COMMIT**

Příkaz pro potvrzení všech změn v databázi.

COMMIT;

Tento příkaz trvale uloží všechny změny, které byly provedeny od posledního potvrzení nebo začátku transakce.

**3**

Při zapnutí počítače probíhá proces známý jako **Power-On Self-Test** (POST), který slouží ke kontrole a inicializaci technického vybavení počítače. Tento test je prováděn BIOSem (Basic Input/Output System) nebo UEFI (Unified Extensible Firmware Interface), což jsou základní systémy uložené na základní desce počítače. POST je kritickou součástí spuštění počítače, protože ověřuje, zda je hardware připraven ke spuštění operačního systému.

### Průběh a kroky POST

1. **Kontrola základního hardwaru**:
   1. Po zapnutí počítače BIOS/UEFI ihned zkontroluje základní hardware, jako je procesor a paměť (RAM). Tyto komponenty musí fungovat správně, aby počítač mohl pokračovat v inicializaci.
   2. POST nejprve provede kontrolu integrity samotného BIOSu/UEFI a inicializuje kontrolu základních systémových komponent.
2. **Test paměti (RAM)**:
   1. Následuje test operační paměti, při kterém POST ověří, zda jsou paměťové moduly připojeny a funkční. Obvykle se otestuje schopnost RAM správně zapisovat a číst data.
   2. Pokud je v paměti problém, zobrazí se chybová zpráva nebo sérií zvukových signálů (pípání), které označují konkrétní problém.
3. **Kontrola grafické karty a zobrazovacího zařízení**:
   1. Pokud počítač projde základní kontrolou paměti, POST inicializuje grafickou kartu a ověří, zda je možné zobrazit výstup na monitoru.
   2. Pokud grafická karta nefunguje správně, POST může zastavit spuštění a zobrazit chybovou zprávu, nebo generovat chybové signály.
4. **Kontrola připojených zařízení**:
   1. BIOS/UEFI zkontroluje připojení dalších základních komponent, jako jsou pevné disky (HDD nebo SSD), optické mechaniky, klávesnice a myš.
   2. Každé zařízení, které je důležité pro spuštění systému, je zkontrolováno. Například, pokud BIOS/UEFI nenajde žádný pevný disk, zobrazí se chybová zpráva.
5. **Kontrola základních vstupně-výstupních portů**:
   1. POST ověřuje funkčnost vstupně-výstupních portů, jako jsou USB porty a další externí konektory. To zajišťuje, že k počítači lze připojit periferní zařízení.
6. **Zobrazení zprávy a přístup k nastavení BIOS/UEFI**:
   1. Pokud POST zjistí problém, zastaví spuštění počítače a může zobrazit chybovou zprávu nebo sekvenci zvukových signálů. U modernějších systémů se také může zobrazit výzva k přístupu do BIOS/UEFI pomocí kláves jako F2, DEL nebo ESC.
7. **Inicializace zavádění operačního systému**:
   1. Pokud všechny kontroly proběhnou bez problémů, POST předá řízení operačnímu systému. BIOS/UEFI nalezne zaváděcí zařízení, obvykle pevný disk nebo SSD s nainstalovaným systémem, a zahájí proces zavádění operačního systému.

### Signalizace chyb při POST

POST používá různé způsoby signalizace, pokud při kontrole hardwaru dojde k problému:

* **Chybová pípnutí (beep codes)**: Krátká série pípnutí označuje konkrétní problém s hardwarem (různé kódy závisí na výrobci BIOSu).
* **Chybové zprávy na obrazovce**: Pokud je funkční grafická karta, chybové hlášení se zobrazí na monitoru.
* **Chybové LED diody**: U některých základních desek jsou diody, které při selhání POST indikují závadu na konkrétní komponentě.

POST je tedy základní proces, který předchází spuštění operačního systému, a jeho účelem je zajistit, že je počítač ve stavu, kdy může bez problémů pracovat.