

Uvažujte relace $student(\underline{učo}, jméno, adresa)$ a $zápis(\underline{učo}, \underline{pkód}, ukončení)$.

Pro výraz v relační algebře

- uved'te příklad obou relací takový, že bude obsahovat alespoň jednoho studenta, který nemá nic zapsaného; (1b)
- na tomto příkladu proveďte vyhodnocení výrazu $student \bowtie zápis$ a napište výsledek; (2b)
- pomocí základních operací relační algebry a operace přirozeného spojení ý výraz $student \bowtie zápis$ definujte. zapsi (3b)

V kontextu databázových systémů stručně a věcně popište:

- co je to transakce; (2b) 25.
- vlastnosti transakčního zpracování. (4b) 3, 5, 10, 11

Je dán B^+ -strom s právě dvěma listovými uzly, který organizuje následujících 8 hodnot: 5, 7, 10, 12, 13, 21, 22, 31. Předpokládejte, že umístění jiné hodnoty již není možné bez reorganizace stromu. Nakreslete

- tento strom a (4b)
- jeho podobu po vložení klíče s hodnotou 11. (2b)

Pozn.: U všech uzlů vždy vyznačte i neobsazená místa.

Uved'te alespoň tři typy integritních omezení používaných v relačních databázových systémech, stručně je popište a uveďte vhodný příklad.

V jazyce SQL napište příkaz pro vytvoření pohledu (VIEW), který na relaci $učitel(\underline{id}, jméno, plat, rok_narození)$:

- skryje atribut $plat$; (3b)
- přidá atribut $věk$ učitele. (3b)

Pozn.: aktuální $letoPočet$ vrací funkce $year(current_date)$. Konkrétní datum narození zanedbejte, tedy věk učitele považujeme za stejný po celý rok.

Navrhněte relační model (tedy vytvořte tabulky) pro knihovnu, která eviduje knihy (název, autor, žánr, isbn, vydavatel), exempláře knihy (opotřebení, datum pořízení) a čtenáře (jméno, adresa). Uvažujte, že

- kniha může mít více exemplářů,
- čtenář si půjčuje exempláře knih a
- konkrétní exemplář může být vypůjčen nejvýše jednomu čtenáři.

Historická data neuvažujte, navržené tabulky mají sloužit pro evidenci aktuálního stavu knihovny.

Pozn.: Vždy uvádějte i primární klíče, případně vhodné atributy pro ně přidejte.

Uvažujte relace *produkt*(produkt_id, *název*, *cena_za_ks*) a *prodej*(produkt_id, měsíc, rok, *počet_prodaných_ks*), které evidují produkty a objemy jejich prodejů v jednotlivých měsících (např. produktů s id 3 se v 11. měsíci roku 2014 prodalo 250 kusů). Zformulujte SQL příkazy, které vrátí:

- a. pro každý produkt jeho id a celkový počet prodaných kusů tohoto produktu; (2b)
- b. celkové tržby za všechny produkty v prosinci 2014; (2b)
- c. id, názvy a ceny produktů, které se v roce 2014 ani jednou neprodaly (tj. nemají v relaci *prodej* záznam s rokem 2014). (2b)

Uvažujte relace *student*(učo, *jméno*, *příjmení*) a *zápis*(učo, pkód, *ukončení*).
Popište fungování algoritmu hash-join při vyhodnocování výrazu $student \bowtie zápis$.

P