WITH CHECK OPTION

**V  databázi si vytvoříme následující zkušební tabulku a doplňme ji daty:**

CREATE TABLE IF NOT EXISTS `lide` (

 `id` int(11) NOT NULL AUTO\_INCREMENT,

 `jmeno` varchar(30) COLLATE utf8\_czech\_ci NOT NULL,

 `prijmeni` varchar(30) COLLATE utf8\_czech\_ci NOT NULL,

 `rok` mediumint(9) NOT NULL,

 PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci AUTO\_INCREMENT=9 ;

INSERT INTO `lide` (`id`, `jmeno`, `prijmeni`, `rok`) VALUES

(1, 'Jan', 'Novák', 1983),

(2, 'Jiří', 'Bém', 1984),

(3, 'Ludvík', 'Černoch', 1985),

(4, 'Kristýna ', 'Janišová', 1986),

(5, 'Ondřej', 'Malík', 1986),

(6, 'Dominka ', 'Trkalová', 1987),

(7, 'Radovan', 'Matyščák', 1974);

**Dále vytvoříme tyto dva pohledy:**

CREATE OR REPLACE VIEW lide\_pohled\_1

(jmeno, prijmeni, rok)

AS SELECT jmeno, prijmeni, rok FROM lide

**WHERE rok >= 1985**

WITH CHECK OPTION;

**Druhý pohled je odvozený z pohledu prvního:**

CREATE VIEW lide\_pohled\_2

(jmeno, prijmeni, rok)

AS SELECT jmeno, prijmeni, rok FROM lide\_pohled\_1

**WHERE prijmeni LIKE '%r%'**

WITH CASCADED CHECK OPTION;

V druhém pohledu jsme použili klauzuli **WITH CASCADED CHECK OPTION**, která bude hlídat jak pohled samotný, tak pohled odvozený (v případě lide\_pohled\_2 bude hlídána jak podmínka lide\_pohled\_2, tak podmínka lide\_pohled\_1 ze kterého je lide\_pohled\_2 odvozen).

**Zkouška:**

Prostřednictvím obou pohledů můžeme data z podkladové tabulky odstraňovat. **DELETE.**

Prostřednictvím obou pohledů můžeme data aktualizovat i vkládat **UPDATE** , **INSERT**.

**ALE**

**Prostřednictvím lide\_pohled\_1 lze vkládat nebo aktualizovat roky narození pouze rovny nebo vyšší 1985, jak definuje podmínka WHERE.**

**Prostřednictvím lide\_pohled\_2 lze vkládat pouze příjmení obsahující písmeno „r“ a zároveň rok narození musí být roven nebo větší 1985, aby byla splněna také podmínka lide\_pohled\_1 z něhož je lide\_pohled\_2 odvozen.**

*Vyzkoušejte*