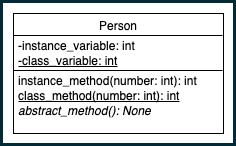
**ZÁKLADNÉ PRVKY CLASS DIAGRAM-u** - súčasť UML **(Unified Modeling Language)**

**Class specification -** špepcifikácia triedy \* - označuje viacero

****

* je privátna, **+** je verejná \_\_\_\_\_\_\_\_\_\_- označuje class variable
* Názov triedy. Začína veľkým písmenom a ak je abstraktná je to *Italic-om*
* Inštančné premenné a ich očakávaný datový typ navratovej hodnoty je int
* Premenné triedy a ich očakávaný typ. Ak je **–** resp. podčiarknutý ide o **private**
* Inštančná metóda s očakávaním jedného parametra správania typu int
* Metóda triedy s jedným parametrom typu int očakávajúca vrátenie hod. int
* Abstraktná metóda vo fonte *Italic* s premennými ktoré nie sú zvonka prístupné a metóda nič nevráti (None) Používa sa tu **enkapsulácia**

názvy premenných s – t.j.

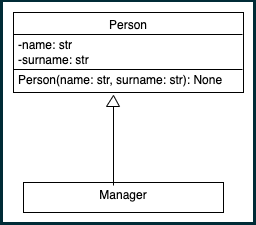
nie sú z vonku prístupne

(enkapsulacia) očakávaný dátový typ návrat. hod. je int

označenie class variable

parametre metódy správania

**Generalization –** dedenie



* Trieda Person
* Trieda Osoba má inštančné parametre name a surname ktoré sú privat typu string
* Trieda Person má konštruktor Person ktorý nič nevracia ( None) s atribútmi
* Trieda Manager je dcerou triedy Person a z nej dedí čo ukazuje smer šípky

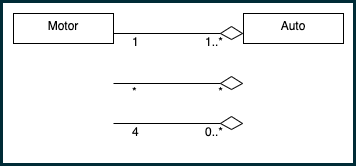
Inštančné atribúty ak by pri nich nebolo žiadne znamienko tak je to ako + t.j. public čiže zvonka prístupné

Konštruktor ktorého názov je ako názov triedy

Neočakáva sa žiadna hodnota

Nemá nič špecifikované

**Agregácia –** spojenie medzi triedami. Používa sa vtedy ak dané objekty vedia existovať aj bez seba resp. ak jeden objekt obsahuje druhý

****

Motor vie existovať aj bez auta a auto vie existovať bez motora.

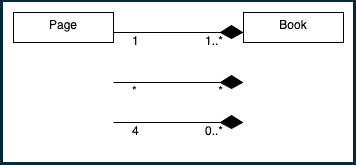
Čísla vyjadrujú početnosťi

* V jednom alebo vo viacerých autách sa nachádza jeden Motor **alebo (..)** viacero Aut má práve jeden Motor. Vzťah tried nie je podmienený a

tak existuje aj spojenie pri nereálnych vzťahoch (viac aut ma 1 a ten istý motor)

* Viacero Motorov sa môže nachádzať vo viacero Autach
* Práve 4 Motory sa nachádzajú v nula(0) autách alebo vo viacerých autách t.j. aj bez Motora môže **existovať** Auto a chceme tento prípad zohladniť

**Composition –** skladanie s existenčnou závislosťou t.j. používa sa vtedy ak objekty nevedia (nemôžu) existovať bez seba

****

Kniha nevie existovať bez strán resp. každá kniha musí mať strany. Ináč to neni kniha ale napr. obal.

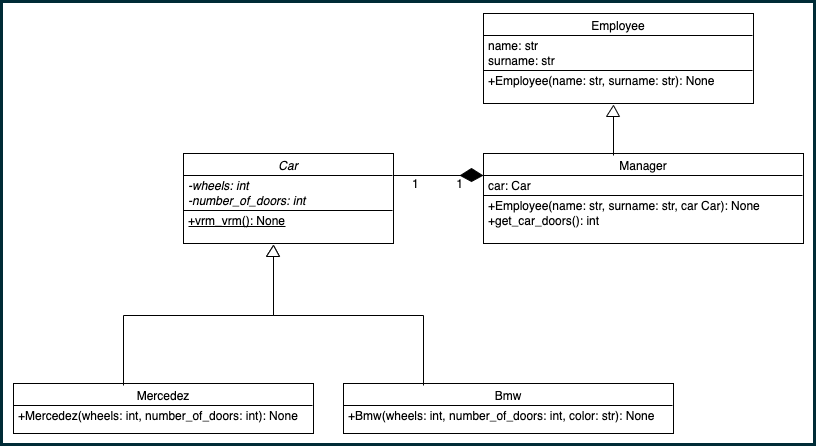
* Kniha nemôže existovať bez jednej strany resp. bez Book-u **neexistuje** žiadna Page t.j. kniha bez strán nemá zmysel alebo (..)Viacero Book-ov má práve jednu Page
* Viacero Book má viacero Page resp. viacero strán má viacero kníh
* Práve 4 stránky sa nachádzaju v nula (0) alebo viacerých knihách

**\*** - označuje viacero

**modul** **ľudia**

je vytvorený v SW **www.umletino.com**

**Príklad použitia:**

****

chýba tu konštruktor

s 2 parametrami ktoré sú

uvedené nižšie

premenné triedy resp.

parametre konštruktora

metóda ktorá môže byť static alebo class

(ak iba metódu použijem a nechcem s ňou pracovať tak ju použijemeako static)

Konštruktor parametre typu int očakávaná hodnota

kvôli parametru **color** musíme zmeniť konštruktor rodiča pomocou prík. **super**

tu je chyba !!! Miesto Employee ma byť **Manager**

public atribut

funkcia ktorá vráti počet

dverí v aute ale kedže number\_of\_dors je privat v triede Car použije sa getter a setter. Očakávanou

návratovou hod. funkcie Manager je None

premenné triedy s očakávaným typom str

metóda triedy s atribútmi

premenných a očakávanou

návratovou hod. None

**modul** **autá**