a++ nebo a-- je postfixová operace, což znamená, že hodnota a se po vyhodnocení výrazu změní.

++a nebo --a je operace s prefixem, což znamená, že hodnota a se před vyhodnocením výrazu změní.

**Object** - V programu reprezentuje reálnou věc. Objekt data seskupuje v paměti. Každý objekt má své vlastní [atributy](https://cs.wikipedia.org/wiki/Atribut) a [metody](https://cs.wikipedia.org/wiki/Metoda_(objektov%C4%9B_orientovan%C3%A9_programov%C3%A1n%C3%AD)) podle vzoru (třídy). Atributy definující objekt však nemusí mít nutně pevné hodnoty. Metody s daty pracují.

**Třída** - Vzor podle kterého se vytváří objekt v paměti. Dá se říct, že je to základní prvek OOP sloužící jako předpis pro vznik objektu jako instance třídy. Třída definuje vlastnosti (atributy) a metody objektů, které při běhu programu vzniknou. Hodnoty vlastností se mohou u jednotlivých objektů odlišovat, na každém objektu, který vznik z jedné třídy, můžeme zavolat pomoci shodnou sadou metod.

**Konstruktor** - Druh metody, která se zavolá k okamžiku vytváření nové instance. Jedná se o metodu, která sestaví objekt/inicializuje objekt. při inicializaci, tj. vzniku objektu, musí být zadány hodnoty atributů jméno a věk

**Getter/Setter** - Pomocné metody pro získání nebo nastavování hodnot v objektu. Složí pro kontrolu/validaci nastavovaných dat. Jsou to metody, které bude možné na instanci - objektu - použít (zavolat)

**Static/non-static** - Statické metody jsou vázané na třídu(static říká, že metoda nepotřebuje pro své fungování žádný konkrétní existující objekt, s nímž by pracovala, To přesně potřebujeme u main, neboť žádný objekt dosud nemáme). Nestatické metody jsou vázané na objekt.

**Instance** – Je objekt. To samé jako object => instance == object

**Modifikátory přístupu:**

| **Je dostupné z:** | **Class** | **Package** | **Potomci** | **Všichni** |
| --- | --- | --- | --- | --- |
| Public | Ano | Ano | Ano | Ano |
| Protected | Ano | Ano | Ano | Ne |
| Bez modidikátoru | Ano | Ano | Ne | Ne |
| Private | Ano | Ne | Ne | Ne |

**Rekurze** – Metoda, která volá sama sebe. Každá rekurze by měla mít podmínku ukončení.   
Jednoduchá implementace metod, přirozenější chování, špatná čitelnost kódu, složité ladění, pokud chyba v kódu

Statická inicializace - Je provedena když je odkazováno na třídu

Inicializace objektu - Provádí se když se vytváří nová instance třídy

Zavolání kontruktoru - První metoda, která se zavolá když objekt vzniká. Má za úkol objekt inicializovat.

Zapouzdření

Dědičnost

Polymorphismus