Object orientede proramlaşdırmada (OOP) öncədə qeyd etdiyim kimi proqramlama obyektlər üzərində aparılır və proqramlayan zaman istifadə etdəcəyimiz obyektlər bir başa kodlaşdırdığımız hissədə funksiyalar və dəyərlər yazaraq yox, əvvəlcədən classlar yaradaraq yartdığımız oyektləri geri cağıraraq istədiyimiz obyekti istədiyimiz yerə yerləşdiririk. OOP-nin aşağıdakı kimi terminləri var:

1. Class - OOP tipli dildə sadəcə obyektlərdən istiafdə olunur və bu obyektlər arasında əlaqə qurmaq mümkündür. Obyektlər öncədən classlardan istifadə edilərək yaradılır və onların ümumi dəyələri verilir, daha sonra harda istifadə ediləcəksə orada həmən class çağırılaraq istədiyimiz özəllikləri onlara verə bilərik. Məsələn: Bir insan silueti ayaradırıq və onun ümumi dəyərlərin yazırıq (iki əli, ayağı, gözü, qulağı, və bir burnu olması) daha sonra bu classdan istifadə edərək onun rəngini (ağbəniz, qarayanız və s. olması) işini və s. daxil edə biıərik. Classların bir birilə sinxron şəkildə işləmə funksiyaları da var. Classlar obyektlərdən fərqli yaddaşda xüsusi yer tutmur, yalnız obyekt yaratdıqda classa nə qədər dəyər vermişiksə obyekt də ona uyğun olaraq yaddaşda yer tutacaq. Classların private (yalnız həmin classa xas dəyər), protected (classla əlaqəli classların da istifadə edə biıəcəyi dəyərlər), və public (bütün classlarda istifadə oluna bilən) dəyərləri olur.
2. Object – obyektlər classlardan yaradılır. Obyektlər classlardan fərqli həcmə malikdir və belə demək mümüknsə canlıdır. İnsan classında yaratdığmız ümumi dəyərlər məsələn avropalı və asiyalı insan yaratmaq istəsək yenidən insan classı çağırılacaq lakin onlara xas olan dəyərlər giriləcək məsələn, asiyalının rəngi qarayanız və gözlərinin qıyıq olması, boylarınmın isə qısa, avropalı insanda isə nisbətən ağbəniz və ucaboy olması və s. dəyərlər. OoP proqramlamada obyektlər və onların qarşılıqlı təsiri önəmlidir.
3. Abstraction – dedikdə yaratdığımız ümumi classa sonradan başqa yerdə istifadə etmək üçün onu çağırıb yalnız ona xas dəyərlərin ona verilməsidir, İnsan misalında olduğu kimi avropalı və asiyalı qarşılaşdırılması abstractiona aiddir.
4. Encapsulation – paketləmə çənasının verir və classı yaradan metod və xüsusiyyətləri reallaşdırma yöntəmidir və bu calssı istiafdə edəcək istifadəçilərdən gizlənməsidir. Məlumat və məlumatın üzərində işləyən kod birlikdə obyekti yaradır. Ümumi obyekti yaradan classın detalları proqramın digər hissələri üçün gizli qalır. Hər hansı obyekt classını reallaşmasını dəyiştirsək, yəni eyni class və metod olmaq şərtilə fərqli proqramlaşırma texnikası ilə reallaşdırsaq, bu classın görünən hissəsi dəyişmirsə, classı yenidən kodlamağa ehtiyac yoxdur.
5. İnheritance (miras almaq) - obyekt yönümlü proqramlamada obyekt, ümumilikdə bir obyekt classına aid varlıqdır (instance). Məsələn, X insan, insan classının varlığıdır. Obyekt class-dan subclass yarada bilirsə, çoxalma xasiyyəti vardır. Məsələn insan classı canlı classının subclassıdır. Subcalassı yaradılan class super, base və ya parent class adlanır. Subclass baseclassdan ona xas xüsusiyyətlər alırsa bu inheritance adlanır. Məsələn insan clasında valideynləri base class kimi yaratsaq onlardan yaranan uşaq classı onlara xas xüsusiyyətləri miraas alacaq. Subclass bir üst classın özəllik və metodlarını olduğu kimi götürür.
6. Polymorphism – fərqli obyektlərin eyni mesaja və ya hadisəyə fərqli cavab verə bilmə xüsusiyyətidir. Hər class öz metodlarını encapsulation etdiyi və digər classlar gizli qaldığı üçün eyni metodlar başqa classlarda da eyni adda yaradıla bilər. Buna görə də yaradılan hər yeni class eyni çağırışa fərqli cavab verəcəkdir. Maşın adlı bir baseclassımız olsun , bu classdan Mercedes, VV, Porsche adlı 3 subclass yaradaq. Bu classların hər biri öz xüsusiyyətini ekrana verməsi üçün business adlı metoda sahib ola bilər. Hər subclass üçün eyni metod olmasına baxmayaraq, bu classları işlətmək üçün göndərdiyimiz mesajdan hər bir class üçün fərqli cavab ortaya çıxacaq (məsələn: Mercedes – S class, VV- Pantheon, Porsche - Panamera)

OOP proqramlamada classların çoxalma funksiyası olduğuna görə obyekt iyerarxiyası yaranma imkanı vardır