



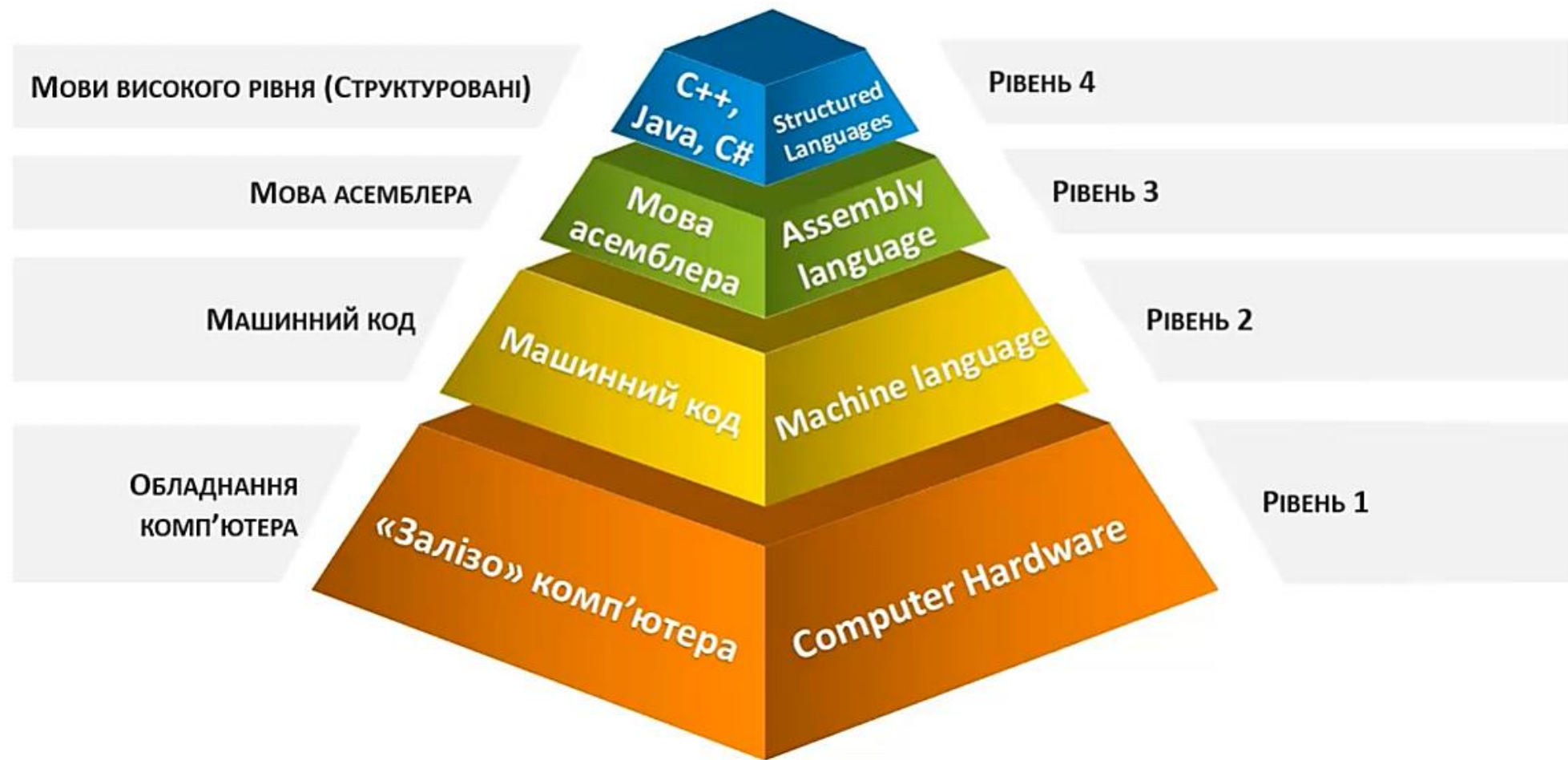
Лекція 1

ВСТУП ДО ОБ'ЄКТНО- ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ (ООП)

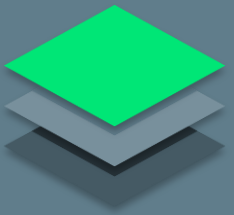


1. Об'єктно-орієнтований підхід
2. Основні поняття ООП
3. Принципи ООП

1. Об'єктно-орієнтований підхід



1. Об'єктно-орієнтований підхід



Машинний код

(Machine language)

```
10111001 00000000
11010010 10100001
00000100 00000000
10001001 00000000
00001110 10001011
00000000 00011110
00000000 00000010
10111001 00000000
11100001 00000011
00010000 11000011
10001001 10100011
00001110 00000100
00000010 00000000
```

(Це програма, яка додає числа 1234 і 4321. Права колонка є продовженням лівої)

Мова асемблера

(Assembly language)

```
MOV CX,1234      ;store 1234 in register CX, and then
MOV DS:[0],CX    ;transfer it to memory location DS:[0]

MOV CX,4321      ;store 4321 in register CX, and then
MOV DS:[2],CX    ;transfer it to memory location DS:[2]

MOV AX,DS:[0]    ;move variables stored in memory at
MOV BX,DS:[2]    ;DS:[0] and DS:[2] into AX & BX

ADD AX,BX        ;add AX and BX, store sum in AX

MOV DS:[4],AX    ;move the sum into memory at DS:[4]
```

(Це програма, яка додає числа 1234 і 4321)

Мова високого рівня

(High-level language)

```
int a=1234;
int b=4321;
int c = a + b;
```

(Це програма, яка додає числа 1234 і 4321)

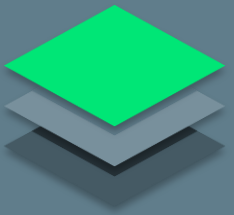
1. Об'єктно-орієнтований підхід



Парадигма програмування (методологія, підхід до програмування) – це система ідей і понять, які визначають **стиль** написання комп'ютерних програм.



1. Об'єктно-орієнтований підхід



Парадигми програмування

Декларативне (описове)
програмування

що шукати? що рахувати?

Функціональне
Lisp, Haskell

Логічне
Prolog

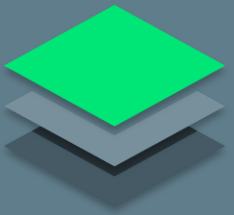
Імперативне (наказове)
програмування

спочатку роби одне, потім інше

Процедурне
Fortran, Algol, Pascal, C

Об'єктно-орієнтоване
C++, Java, C#, Python

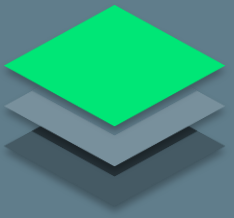
1. Об'єктно-орієнтований підхід



Процедурне програмування зображає програму у вигляді **набору алгоритмів**, для оформлення яких можуть бути застосовані іменовані програмні блоки – **процедури та функції**.

Структурне програмування – це варіант процедурного, що використовує три типи структур керування: *послідовне виконання дій, розгалуження і цикл*.

1. Об'єктно-орієнтований підхід



Процедурне програмування

- Функціональність понад усе!
 - Дані другорядні
- Весь код в одному файлі
- Незручний для великих програм

1. Об'єктно-орієнтований підхід



Об'єктно-орієнтоване програмування (ООП) – це методологія програмування, заснована на записі програми у вигляді сукупності **об'єктів**, кожний з яких є екземпляром певного **класу**, а класи утворюють **ієрархію спадкування**.

1. Об'єктно-орієнтований підхід



Об'єктно-орієнтоване програмування:

- ✓ моделювання об'єктів реального світу;
- ✓ типи даних, які розроблені користувачем;
- ✓ закриття деталей реалізації;
- ✓ багаторазове використання програмного коду;
- ✓ інтерпретація викликів функцій на етапі виконання.

2. Основні поняття ООП

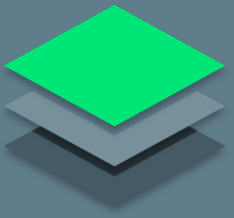


Клас (class)

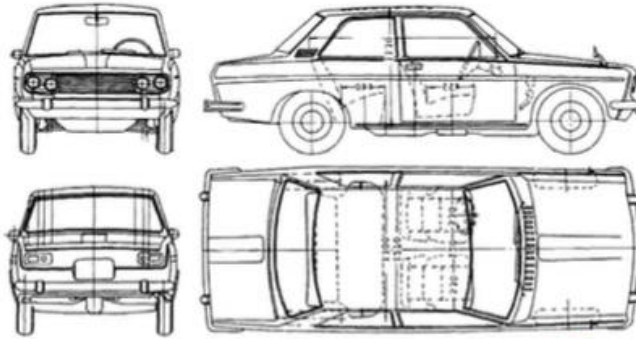
1. 4 колеса
2. 1 кермо
3. Двигун
4. Фари
5. Може їхати (не може літати чи плавати під водою)
6. ...



2. Основні поняття ООП



Клас (class)



Об'єкт класу "Автомобіль"



Об'єкт класу "Автомобіль"



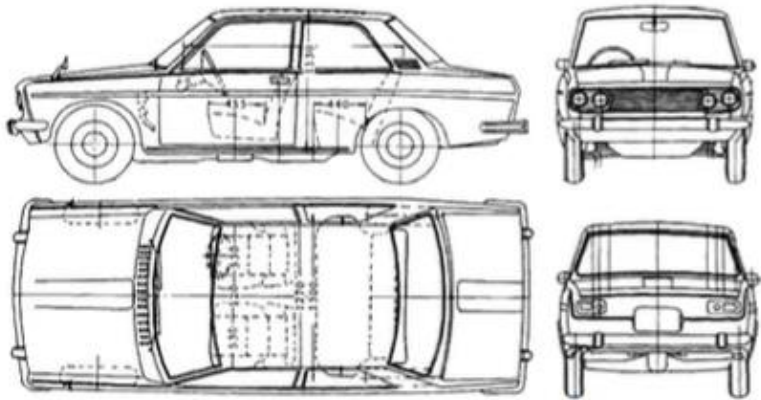
Об'єкт класу "Автомобіль"

2. Основні поняття ООП



Клас (class)

Концептуальний опис об'єкта (шаблон)



Клас: Автомобіль



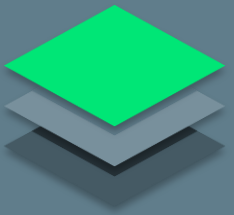
Об'єкт (object)

Екземпляр класу (instance of a class)



Об'єкт Mercedes CLS
(Класу Автомобіль)

2. Основні поняття ООП



Клас (class) – визначений програмістом **тип даних**, що використовується для опису **об'єктів**.

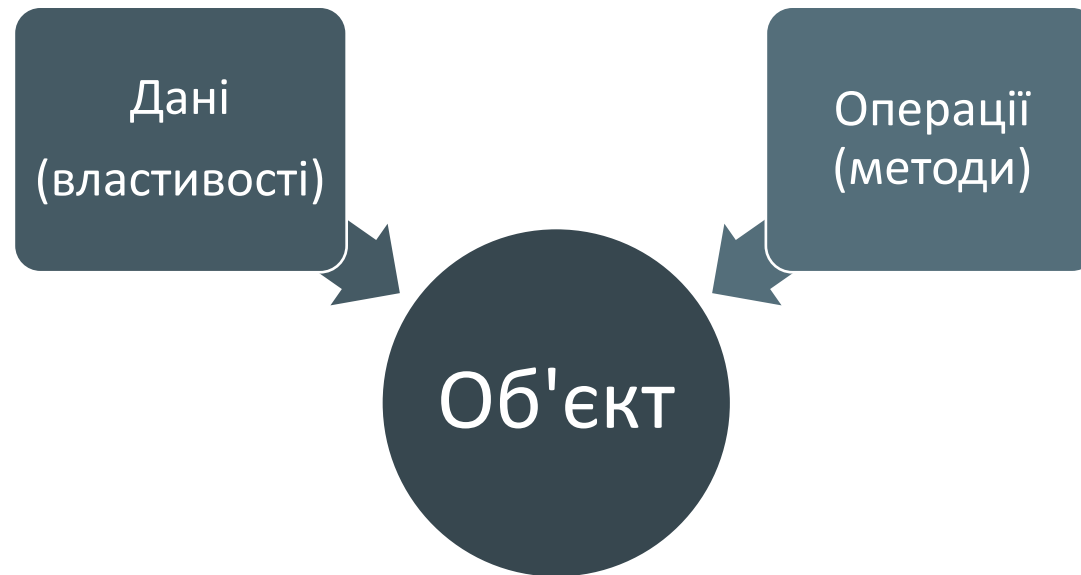
Об'єкт (object) – це структурована змінна типу **клас**, що містить всю інформацію про деякий елемент, який реалізується в програмі.

2. Основні поняття ООП

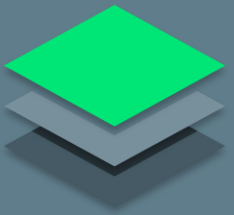


Екземпляр класу (*instance of a class*) – це конкретний об'єкт з набору об'єктів даного класу.

На основі **одного класу** може бути створено **безліч об'єктів**.



2. Основні поняття ООП



Властивості об'єкта (object properties, attributes) – це прості змінні, які характеризують об'єкт.

Стан об'єкта (state) – це перелік всіх властивостей даного об'єкту з поточними значеннями кожної властивості.

2. Основні поняття ООП



Властивості об'єкта (Object properties)

Виробник: **Mercedes**

Модель: **CLS**

Рік випуску: **2017**

Колір: **Червоний**

Ціна: **\$55 000**

Відкинний дах: **ні**

2. Основні поняття ООП



Властивості об'єкта (Object properties)

Виробник: **Ford**

Модель: **Mustang**

Рік випуску: **2016**

Колір: **Жовтий**

Ціна: **\$34 000**

Відкинний дах: **так**

2. Основні поняття ООП

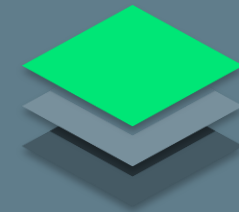


Поведінка (*behavior*) – це набір операцій (методів) об'єкта.

Методи (*methods*) – це функції, що реалізують можливі дії з об'єктом.



3. Принципи ООП



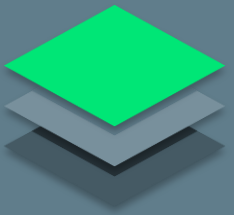
Абстрагування

Інкапсуляція

Поліморфізм

Успадкування

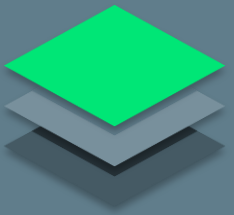
3. Принципи ООП



Абстрагування (*abstraction*) – спосіб виділити набір **значущих характеристик** об'єкта, виключаючи з розгляду незначущі.

Класи є результатом роботи **абстрактного мислення** людини з використанням операцій **групування** та **узагальнення**.

3. Принципи ООП



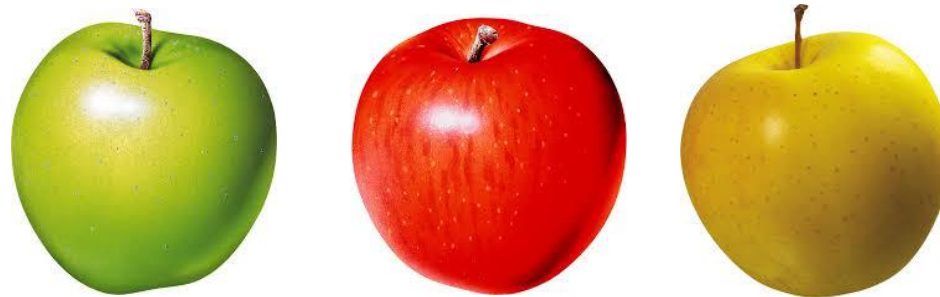
Абстрагування



Клас



Групування



Екземпляр

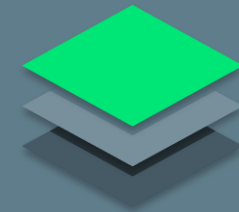
3. Принципи ООП



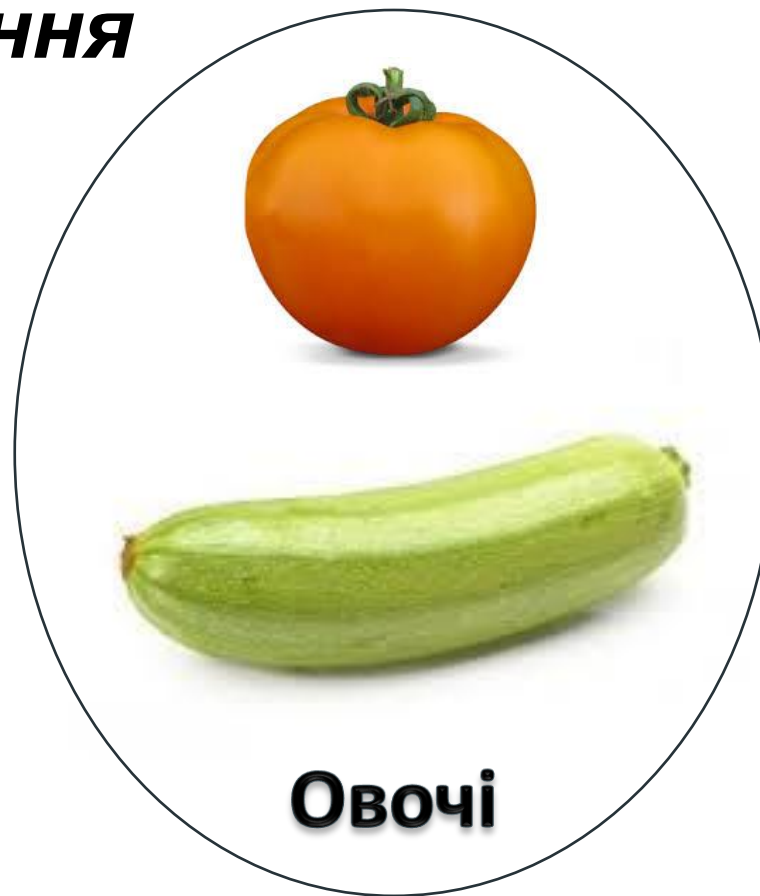
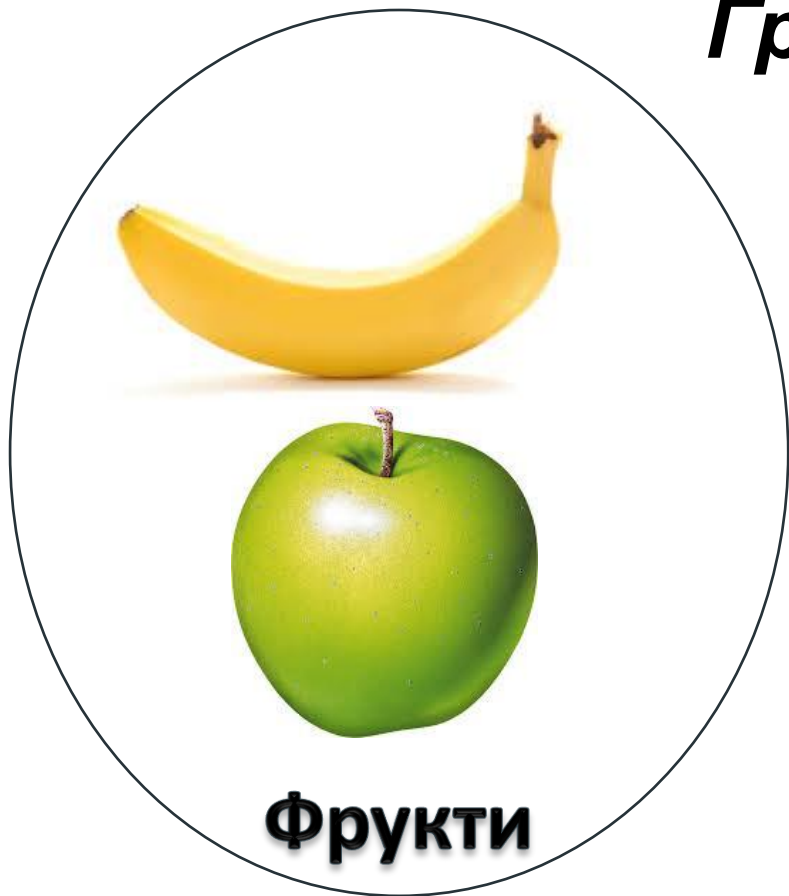
Групування



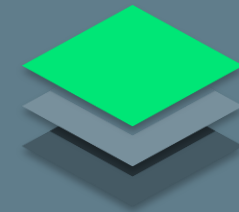
3. Принципи ООП



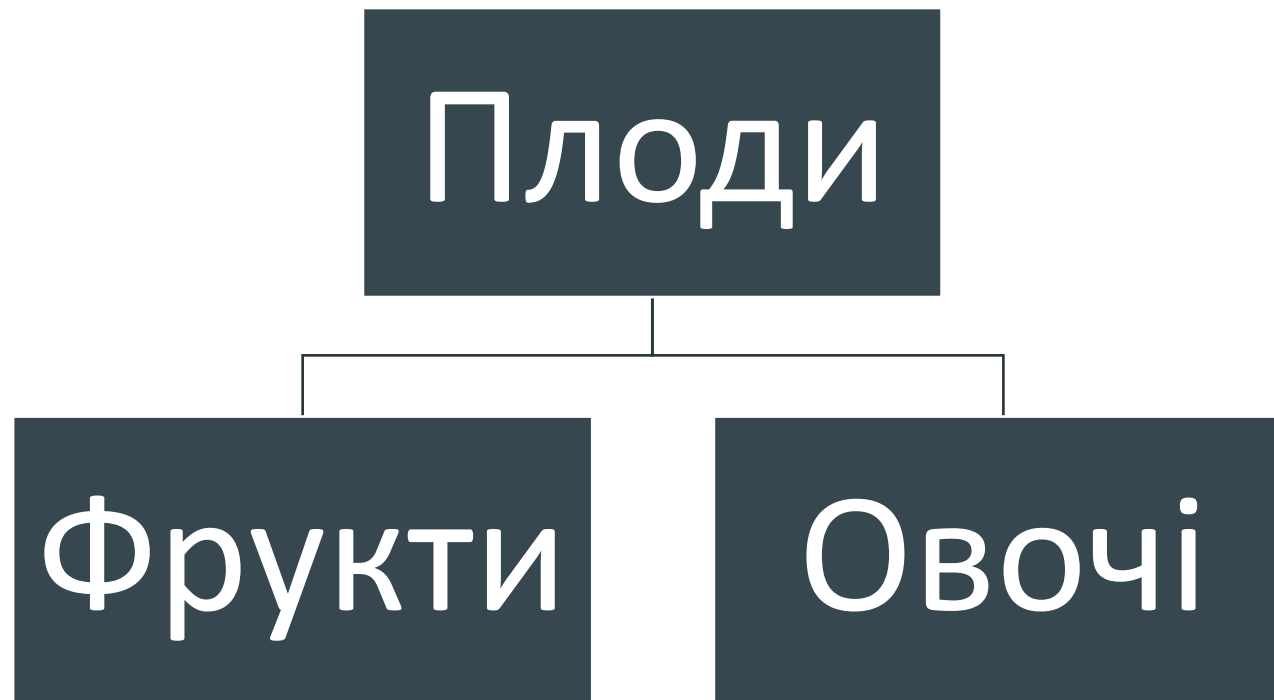
Групування



3. Принципи ООП



Узагальнення



3. Принципи ООП

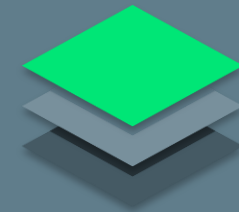


Абстрактне мислення дозволяє провести **декомпозицію** предметної області на набір понять і зав'язків між ними.

Декомпозиція програми передбачає створення модулів.

Модуль (*module*) – це окрема самостійна частина алгоритму (деякий блок), що має свою назву, функціональну цілісність і завершеність.

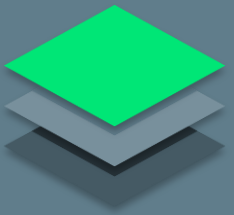
3. Принципи ООП



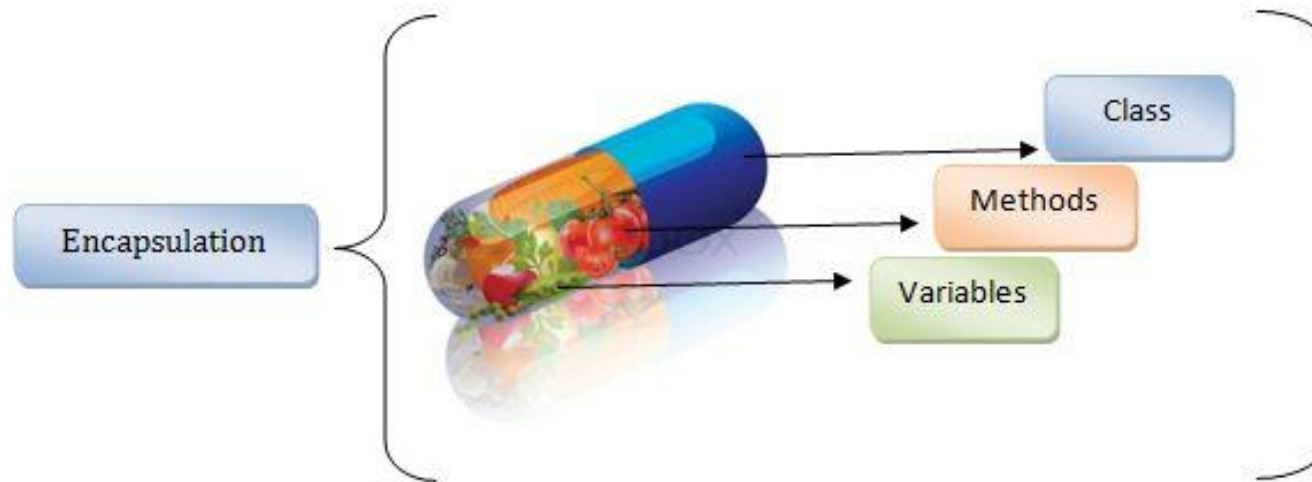
Декомпозиція



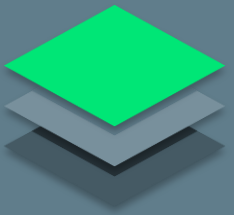
3. Принципи ООП



Інкапсуляція (*encapsulation*) – приховування реалізації класу і відділення його внутрішнього представлення від зовнішнього (інтерфейсу).



3. Принципи ООП

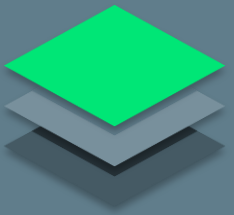


Інтерфейс (interface) – це зовнішній вигляд класу, що виділяє його істотні риси і не показує внутрішній устрій.

Реалізація (implementation) – внутрішнє представлення класу, включаючи особливості його поведінки.



3. Принципи ООП

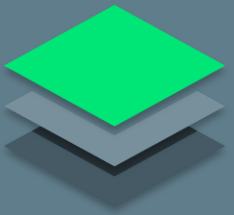


Поліморфізм (polymorphism) – властивість, що дозволяє вирішувати схожі за змістом завдання різними методами.

Поліморфізм полягає в здатності об'єкта **вибирати** потрібний метод в залежності від типу отриманих даних.



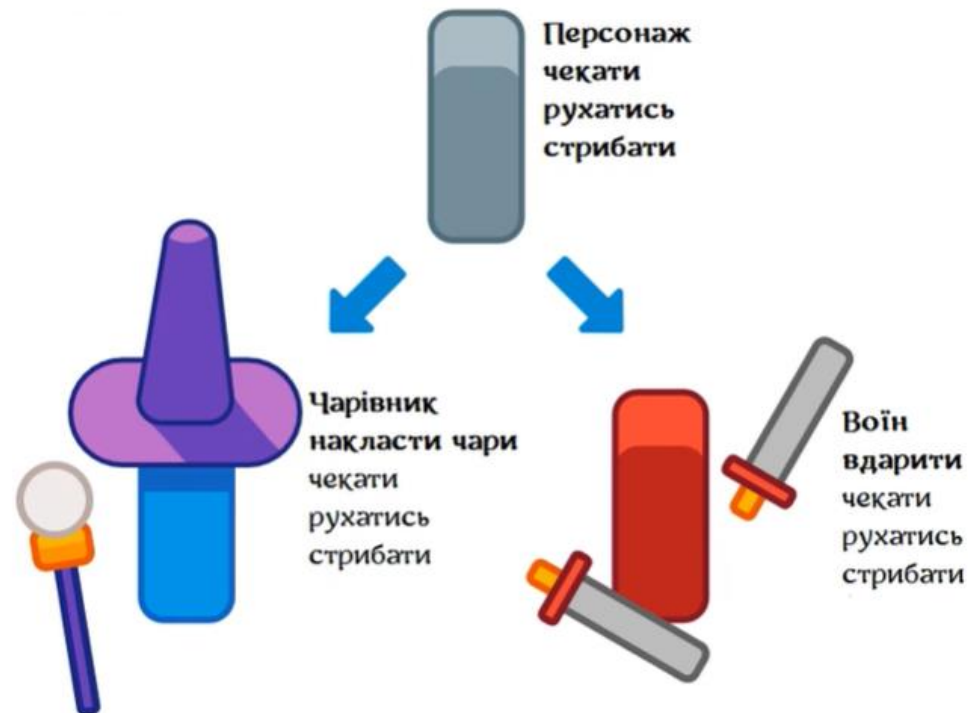
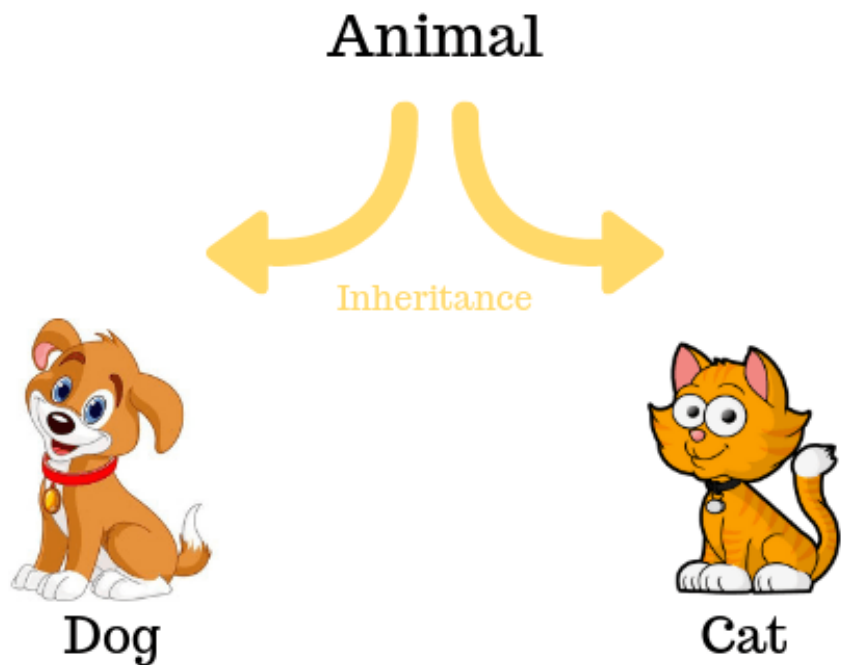
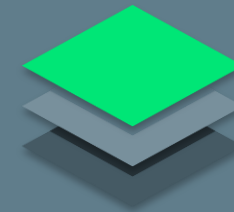
3. Принципи ООП



Успадкування або наслідування (*inheritance*) – співвідношення між класами, коли один клас використовує структурну або функціональну частину іншого класу.

При спадкуванні характеристики одного об'єкта (об'єкта-предка) можуть **передаватися** іншому об'єкту (об'єкту-нащадку) без їх повторного опису.

3. Принципи ООП



Корисні посилання



Що таке ООП

Клас та об'єкт

Методи класу