# **GRASP**

**GRASP** – describes fundamental principles of OOP design

* **G**eneral
* **R**esponsibility
* **A**ssignment
* **S**oftware
* **P**atterns

OOP principles -> GRASP -> GoF

|  |  |  |
| --- | --- | --- |
| ID | PRINCIPLE | DESCRIPTION |
| 1 | **CREATOR** |  |
| 2 | **INFORMATION**  **EXPERT** |  |
| 3 | **LOW COUPLING** |  |
| 4 | **CONTROLLER** |  |
| 5 | **HIGH COHESION** |  |
| 6 | **INDIRECTION** |  |
| 7 | **POLYMORPHISM** |  |
| 8 | **PPROTECTED VARIATIONS** |  |
| 9 | **PURE FABRICATIONS** |  |

# **#1. CREATOR - PRINCIPLE**

[Creator] identifies what class should be responsible for creating a new instance of a class

**What problem solves – who should be responsible for creating object?**

**Answer – class that uses that object should create it**.

Pros: not increase coupling

Cons: can be complicated

# **#2. INFORMATION EXPERT - PRINCIPLE**

**What problem solves – delegation of responsibilities, for example, calculation of total price**?

Answer – calculate/process info should that component that contains that info