## **🪖 OOP Project: IDF Operation – First Strike**

### **📘 Background**

The State of Israel has been engaged in a conflict with the terrorist organization Hamas since 2023. As part of the IDF (Israel Defense Forces), your mission is to build a software simulation of military operations, combining intelligence analysis and strategic air/artillery strikes.

This object-oriented programming project involves modeling real-world military elements such as units, operations, and intelligence, using proper abstraction and encapsulation.

### **🧱 Core Class Definitions (Entities)**

#### **🟦 IDF**

Represents the Israeli military force.

* Properties:  
  + Date of establishment
  + Current commander
  + A collection of strike options (e.g., aircraft, drones, artillery)

#### **🟦 Strike Options**

The IDF can perform attacks using various strike units, each with different capabilities and targets. All strike types share the following:

* A unique name
* Ammunition capacity (number of strikes remaining)
* Fuel supply
* Type of target they are effective against (e.g., buildings, people, vehicles)

**Subtypes:**

* **F16 Fighter Jet**
  + Bomb types: 0.5 ton or 1 ton
  + Effective against: buildings
  + Operated by a pilot
  + Up to 8 strikes available
* **Hermes 460 ("Zik") Drone**
  + Bomb types vary depending on target: personnel or armored vehicles
  + Effective against: people, vehicles
  + Up to 3 strikes available
* **M109 Artillery**
  + Bomb type: explosive shells
  + Effective in: open areas
  + Can strike up to 2–3 targets simultaneously, 40 strikes in total.

#### **🟦 AMAN (Military Intelligence)**

The IDF’s intelligence branch.

* Responsible for generating **Intelligence Messages**:  
  + Linked to a specific terrorist
  + Include the last known location of the terrorist (e.g., "home", "in a car", "outside")
  + Include a timestamp of when the intel was gathered

#### **🟥 Hamas**

Represents the enemy organization.

* Properties:  
  + Date of formation
  + Current commander
  + A list of affiliated terrorists

#### **🟥 Terrorist**

Members of Hamas tracked by intelligence.

* Properties:  
  + Name
  + Rank (1–5 scale, where 5 is the highest)
  + Status (alive or dead)
  + Weapons (may include one or more from: knife, gun, M16, AK47)

### **🧪 Functionality and Simulation Tasks**

#### **✅ Initialization Phase:**

* Create 5–10 random terrorists and assign them to Hamas  
  + Each terrorist should have 1 or more weapons and a random rank
* Initialize the IDF with its 3 types of strike options
* Generate 10–20 random intelligence messages about real terrorists

#### **🎮 Interactive Menu (Commander Console)**

As the head of the IDF strike team, you have access to a strategic control interface with the following actions:

1. **Intel Analysis**
   * Identify the terrorist with the most intelligence reports
2. **Strike Availability**
   * Show all currently available strike units and their remaining capacity
3. **Target Prioritization**
   * Determine the **most dangerous terrorist** based on a **quality rank**:  
     + Weapon points:  
       - Knife = 1
       - Gun = 2
       - M16 / AK47 = 3
     + Quality Score = Rank × Total Weapon Points  
        *(e.g., a terrorist with rank 2 and weapons knife + gun → 2 × (1 + 2) = 6)*
   * Display their name, rank, quality score, weapons, and **latest known location** from intel
4. **Strike Execution**
   * Based on the terrorist's location and type, choose an appropriate strike unit
   * Confirm the strike with:  
     + Time of order
     + Target (terrorist)
     + Ammunition used
     + Officer's name (your name)
     + Related intelligence (latest one)
   * Update the strike option's remaining capacity

## **🧩 Extra Optional Features (Grouped by Category)**

### **📦 Intelligence & Recon Features**

1. **Intelligence Expiry**
   * Each intelligence message is only valid for 24 hours (based on timestamp).
   * Add logic to **filter expired reports** during target prioritization and strike planning.
2. **Confidence Score for Intel**
   * Each intelligence message has a confidence score (1–100).
   * Prioritize intelligence accordingly or discard low-confidence reports.
3. **Intel Source Tracking**
   * Track the source of each report (e.g., drone, undercover agent, cyber unit).
   * Optional: Some strike options can request updated intel from specific sources.

### **💣 Strike Management Features**

1. **Fuel Consumption & Refueling**
   * Each strike reduces fuel.
   * If fuel < threshold, the unit must return to base for refueling (simulate time delay or disable until refueled).
2. **Strike Cooldown**
   * After a strike, units enter a cooldown period (simulate repair, reloading, etc.).
   * Prevent overuse and require diversified strike planning.
3. **Damage Report Generation**
   * After each strike, generate a report:  
     + Was the terrorist eliminated?
     + Collateral damage? (random chance)
     + Intelligence gained post-strike?

### **🧠 Tactical Decision-Making Features**

1. **Risk Assessment Algorithm**
   * Before striking, show a calculated **"Risk Level"**:  
     + Based on terrorist danger level, location exposure, intelligence confidence, and ammo left.
2. **Strike Recommendation System**
   * Based on terrorist type and location, the system suggests the best strike unit.
   * Use polymorphism to match effectiveness (e.g., drone effective against people in cars).
3. **Strike Logs History**
   * Keep a chronological list of all strikes with:  
     + Terrorist name
     + Officer name
     + Weapon used
     + Outcome
     + Remaining ammo

### **🧬 Additional Entity and Hierarchy Extensions**

1. **Add Ground Forces**
   * New strike type: **Ground Unit**
   * Lower ammo, but reusable (cooldown based).
   * Can capture (not kill) targets, enabling interrogation mechanics.
2. **Terrorist Movement**
   * Each terrorist changes location randomly every X minutes (simulation step).
   * Require updated intel before planning a strike.
3. **Strike Unit Inheritance Hierarchy**
   * Introduce a common interface/base class IStrikeUnit.
   * Use polymorphism for methods like CanStrikeTarget, ConsumeAmmo, Report.

### **💻 Simulation Control Features**

1. **Time Advancement System**
   * Add a simulation clock.
   * Let the user advance time by X minutes/hours.
   * Used for refreshing intel, changing locations, cooldown ending, etc.
2. **User Authentication**
   * Ask the user for a name and code before allowing strike authorization.
   * Store officer identity in all action logs.
3. **Mission Outcomes and Victory Conditions**
   * After N strikes or elapsed time, evaluate performance:  
     + % of terrorists eliminated
     + Ammo efficiency
     + Collateral damage (if modeled)

### **🧪 Analytical Tools and Reports**

1. **Weapon Usage Statistics**
   * Track how many times each strike unit type was used.
   * Which weapons were most effective.
2. **Intel Accuracy Tracking**
   * Compare intel against real terrorist locations.
   * Calculate accuracy score per intel source.
3. **Interactive Search**
   * Search for a terrorist by name or weapon.
   * Show all related intel, rank, and known details.