What we can check:

Returns if the tank is firing.

```
bool IsFiring
```

Returns if the tank is destroyed.

```
bool IsDestroyed
```

Returns float value of remaining health.

```
float GetHealthLevel
```

Returns float value of remaining ammo.

```
float GetAmmoLevel
```

Returns float value of remaining fuel.

```
float GetFuelLevel
```

Returns list of friendly bases.

```
List<GameObject> GetMyBases
```

Returns Dictionary(GameObject target, float distance) of visible targets (tanks in TankMain LayerMask).

```
Dictionary<GameObject, float> GetAllTargetTanksFound
```

Returns Dictionary (GameObject consumable, float distance) of visible consumables (consumables in Consumable LayerMask).

```
Dictionary<GameObject, float> GetAllConsumablesFound
```

Returns Dictionary(GameObject base, float distance) of visible enemy bases (bases in Base LayerMask).

```
Dictionary<GameObject, float> GetAllBasesFound
```

What we can use:

Request a path from this to pointInWorld

```
void FindPathToPoint(GameObject pointInWorld)
```

Follow path to a target (GameObject) at speed (value between 0-1)

```
void FollowPathToPoint(GameObject pointInWorld, float normalizedSpeed)
```

Follow path to a random target (GameObject) at speed (value between 0-1)

```
void FollowPathToRandomPoint(float normalizedSpeed)
```

Set another random point.

```
void GenerateRandomPoint()
```

Stop tank

```
void StopTank()
```

Start tank

```
void StartTank()
```

Face turret to pointInWorld

```
void FaceTurretToPoint(Vector3 pointInWorld)
```

Reset turret

```
void ResetTurret()
```

Fire at pointInWorld

```
void FireAtPoint(GameObject pointInWorld)
```