T-34 vs Tiger — Custom Mission Name Checklist

1. Files and Folders Needed

- `Missions/MyMission/YourMissionName/`
  - `Mission.script`
  - `Content.script`
  - `YourMissionNameMission_Strings.script`
- `Resources/`
  - `MissionYourMissionName.rsr`

---

2. Naming Conventions

- `Mission.script`:
  - `class CYourMissionNameMission extends CSPMission`
- `Strings.script`:
  - `class YourMissionNameMission_Strings`
- `.rsr` section:
  - `[MissionYourMissionName]`

---

3. Inside Each File

Mission.script

```
static WString ObjectivesText = YourMissionNameMission_Strings::ObjectivesText;

void CYourMissionNameMission()
{
    CSPMission("CYourMissionNameMission", "YourMissionNameContent");
}
```

Strings Script

```
class YourMissionNameMission_Strings
{
  static WString MissionName = L"Your Mission Name Here";
  static WString ObjectivesText = L"Objective 1\nObjective 2\nObjective 3";
  static WString Objective01 = L"Objective 1";
```

```
    static WString Objective02 = L"Objective 2";
    static WString Objective03 = L"Objective 3";
}
```

MissionYourMissionName.rsr

```
[MissionYourMissionName]
MissionName    = "Your Mission Name Here"
BriefingText   = "Your Briefing Text Here"
ObjectivesText = "Objective 1\nObjective 2\nObjective 3"
```

---

4. MissionsMenu.script

Inside the correct [MissionXXX] section (USSR or Axis campaign), add:

```
{
    "Resources/MissionYourMissionName.rsr",
    new #MissionCSPData<CYourMissionNameMission>(),
    0
},
```

---

5. Final Pre-Launch Checks

- Correct folder paths

- Class names exactly match file names

- MissionName field is NOT missing

- No typos between .script and .rsr

- .rsr file is present in /Resources

- All scripts saved and compiled if needed