T-34 vs Tiger — Custom Mission Name Checklist (Updated)

1. Files and Folders Needed

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

2. Naming Conventions (Critical!)

Purpose	Example
Mission Class	class CYourMissionNameMission extends CSPMission
Strings Class	class YourMissionNameMission_Strings
.rsr Section Header	[MissionYourMissionName]
.rsr File Name	MissionYourMissionName.rsr
Content Script Class	class YourMissionNameContent
Menu Entry Class Call	new #MissionCSPData<CYourMissionNameMission>()

3. Inside Each File

Mission Script (Mission.script)

```
class CYourMissionNameMission extends CSPMission
{
  static WString ObjectivesText =
YourMissionNameMission_Strings::ObjectivesText;

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

Strings Script (YourMissionNameMission_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";
}
```

RSR File (MissionYourMissionName.rsr)

```
[MissionYourMissionName]

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

4. Edit MissionsMenu.script

Find the USSR_Missions or Germany_Missions array and add a new entry:

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

Important:

- Path points to the .rsr file inside Resources/
- Mission class must match your actual mission script class (CYourMissionNameMission)

5. Final Pre-Launch Checklist

✅ Correct folder structure
✅ Class names EXACTLY match to filenames and resource names
✅ MissionName field present in .rsr AND in _Strings script
✅ .rsr file placed inside /Resources/
✅ No typos in any file or class name
✅ Scripts saved correctly (no missing semicolons, correct brackets {})
✅ Editor/engine restarted if needed (reload mission list)

Important Observations

- The .rsr file controls what appears on the main mission list (MissionName).
- The _Strings.script file controls Objectives and Briefing inside the mission.
- If either one is missing, wrong, or broken, you will see TEXT or blank fields.
- Naming consistency is absolutely critical.

Summary

You must have both the .rsr file and the _Strings.script file correctly linked in the Mission.script and MissionsMenu.script to display a proper named, described, and objective-filled mission.