

P3 Code Reference

Warning and Error Code
Examples and Explanations

Developed by Subsimple

Version 2018 December 30
For P3 v1.3.0

Warnings

When a warning criteria is met, a message will be added to the screen.

A warning does not affect P3's operation and is solely implemented to notify the creator of a certain occurrence.

The number of warnings is displayed during P3's ending sequence.

Each warning is associated with a unique identifier which may be used to look up its associated warning criteria in this guide.

Every warning can be configured to be suppressed and ignored.

—

Warnings will come in 2 forms.

1. With line number
2. Without line number

Warnings *with* line numbers will generally be more specific in pinpointing the relative location of the occurrence.

Conversely, warnings *without* line numbers will be more vague and cover multiple lines.

Warning - W0101

Message	Wave X's credits is not a multiple of Y X - Wave Number Y - Configuration No line number shown.
Config	int_warn_credits_multiple Default: 50
Example	Wave // Total credits dropped during this wave: 100 + 17 = 117 { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 TotalCurrency 100 Where spawnbot TFBot { Class Scout Skill Easy } } WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 TotalCurrency 17 Where spawnbot TFBot { Class Pyro Skill Easy } } }
Explanation	The wave in the example drops 100 and 17 credits, totaling 117 credits without bonus. 117 is not a multiple of 50.

Warning - W0102

Message	Total Possible Credits exceeds maximum possible reading of 30000 No line number shown.
Config	bool_warn_credits_gr_30000 Default: true
Example 1	<pre>WaveSchedule { StartingCurrency 99999 // Causes W0102 Wave { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Heavy Skill Easy } } } }</pre>
Example 2	<pre>WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 TotalCurrency 30001 // Causes W0102 Where spawnbot TFBot { Class Scout Skill Easy } }</pre>
Explanation	<p>The maximum possible credits a player can be holding at once cannot exceed 30000.</p> <p>If a player currently holds 30000 credits, the player must spend credits in order to pick up more credits to be held.</p>

Warning - W0103

Message	TotalCurrency value equal to 0 drops nothing
Config	bool_warn_totalcurrency_0 Default: false
Example	WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 TotalCurrency 0 // W0103 Where spawnbot TFBot { Class Scout Skill Easy } }
Explanation	This is a redundant line that does nothing.

Warning - W0201

Message	Invalid Attribute Name found
Config	bool_warn_invalid_item_char_attribute Default: true
Example 1	TFBot { Class Scout Skill Easy Item "Mad Milk" ItemAttributes { ItemName "Mad Milk" "invalid item attribute key" 2 // W0201 } }
Explanation 1	The entered item attribute does not exist according to TF2. Misspelled attribute names typically trigger this warning.
Example 2	TFBot { Class Scout Skill Easy CharacterAttributes { "invalid character attribute key" 2 // W0201 } }
Explanation 2	The entered character attribute does not exist according to TF2. Misspelled attribute names typically trigger this warning.

Warning - W0202

Message	Attribute does nothing
Config	bool_warn_attribute_value_type_scan Default: false Experimental
Example	<pre>TFBot { Class Scout Skill Easy Item "The Shortstop" ItemAttributes { ItemName "The Shortstop" "damage bonus" 1 // W0202 } }</pre>
Explanation	<p>This feature is experimental and is by default OFF.</p> <p>The attribute is given a value that does not alter its stats when compared to stock weapons. The comparison to stock weapons causes widespread unreliability.</p> <p>Affected attribute values may multiply/divide the default value by 1.00 or add/subtract 0 from the default value.</p>

Warning - W0203

Message	N/A
Config	N/A
Example	N/A
Explanation	Warning W0203 is unused.

Warning - W0204

Message	Attribute can only be values 0 or 1
Config	<code>bool_warn_attribute_value_type_scan</code> Default: false Experimental
Example	<pre>TFBot { Class Heavy Skill Easy Item "The Huo Long Heatmaker" ItemAttributes { ItemName "The Huo Long Heatmaker" "crit vs burning players" 99 // W0204 } }</pre>
Explanation	<p>This feature is experimental and is by default OFF.</p> <p>The attribute's type is <i>value_is_or</i>. <i>Value_is_or</i> attributes are only defined as true or false with the values 1 or 0, respectively.</p>

Warning - W0205

Message	TFBot Health is not a multiple of X X - Configuration
Config	int_tank_health_multiple Default: 5
Example	TFBot { Class Scout Skill Easy Health 123 // W0205 }
Explanation	123 is not a multiple of 5.

Warning - W0206

Message	Tank Health is not a multiple of X X - Configuration
Config	int_tank_health_multiple Default: 500
Example	<pre>Tank { Name TankBoss Health 789 // W0206 Speed 75 StartingPathTrackNode boss_path_1 OnBombDroppedOutput { Target boss_deploy_relay Action Trigger } }</pre>
Explanation	789 is not a multiple of 500.

Warning - W0207

Message	Tank Health exceeds maximum warning [X] X - Configuration
Config	int_tank_warn_maximum Default: 100000
Example	<pre>Tank { Name TankBoss Health 999999 // W0207 Speed 75 StartingPathTrackNode boss_path_1 OnBombDroppedOutput { Target boss_deploy_relay Action Trigger } }</pre>
Explanation	999999 is greater than 100000.

Warning - W0208

Message	Tank Health is below minimum warning [X] X - Configuration
Config	int_tank_warn_minimum Default: 10000
Example	<pre>Tank { Name TankBoss Health 1234 // W0208 Speed 75 StartingPathTrackNode boss_path_1 OnBombDroppedOutput { Target boss_deploy_relay Action Trigger } }</pre>
Explanation	1234 is less than 10000

Warning - W0209

Message	Invalid TF2 Item Name
Config	bool_warn_invalid_item_name Default: true
Example	<pre>TFBot { Class Scout Skill Easy Item "invalid item" // W0209 }</pre>
Explanation	The specified item does not exist within TF2's item list. Misspelled names typically trigger this warning.

Warning - W0210

Message	TFBot does not have item: X X - ItemName value
Config	bool_warn_tfbot_missing_item Default: true
Example 1	<pre>TFBot { Class Scout Skill Easy ItemAttributes { ItemName "Mad Milk" // W0210 "effect bar recharge rate increased" 2 } }</pre>
Explanation 1	A modification to an item the TFBot does not possess.
Example 2	<pre>TFBot { Class Scout Skill Easy ItemAttributes { ItemName "The Shortstop" // W0210 False Positive "damage bonus" 2 } Item "The Shortstop" }</pre>
Explanation 2	<p>This is a false positive and works perfectly fine in game.</p> <p>A modification to the item was given to the item in a line above the line where the item is given to the TFBot.</p>

Warning - W0211

Message	Template does not exist: X X - Template value
Config	bool_warn_bad_template Default: true
Example	<pre>TFBot { Class Scout Skill Easy Template template_does_not_exist // W0211 }</pre>
Explanation	The specified template value was never defined earlier.

Warning - W0212

Message	Bot name cannot display symbol: '%'
Config	bool_warn_tfbot_bad_character Default: true
Example	<pre>TFBot { Name "Percent % Scout" // W0212 Class Scout Skill Easy }</pre>
Explanation	Shows in game as "Percent Scout" The symbol '%' is a reserved character that defines variables in TF2's hud.

Warning - W0213

Message	Tank not named 'TankBoss' does not explode on deployment
Config	bool_warn_tank_name_tankboss Default: false
Example	<pre>Tank { Name "BadTankName" // W0213 Health 25000 Speed 75 StartingPathTrackNode boss_path_1 OnBombDroppedOutput { Target boss_deploy_relay Action Trigger } }</pre>
Explanation	<p>This is a minor cosmetic warning and is by default OFF.</p> <p>On successful deployment, the tank does not suicide with the detonation of the hatch.</p> <p>The tank must be named <i>TankBoss</i> to be able to suicide with the hatch.</p>

Warning - W0301

Message	WaitForAll* name does not exist: X X - WaitForAllSpawned value or WaitForAllDead value
Config	bool_warn_wait_for_all_not_found Default: true
Example 1	WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot WaitForAllDead "wavespawn_doesnt_exist" // W0301 TFBot { Class Scout Skill Easy } }
Explanation 1	WaveSpawn waits for a nonexistent WaveSpawn's bots to all spawn before spawning. This is a common softlock for many missions.
Example 2	WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot WaitForAllSpawned "wavespawn_doesnt_exist" // W0301 TFBot { Class Scout Skill Easy } }
Explanation 2	WaveSpawn waits for a nonexistent WaveSpawn's bots to all die before spawning. This is a common softlock for many missions.

Warning - W0302

Message	Possible premature end of WaveSchedule detected near X X - Line that ends WaveSchedule, typically a '}'
Config	bool_warn_early_end_wave_schedule Default: false Experimental
Example	N/A
Explanation	<p>This feature is experimental and is by default OFF.</p> <p>Having too many closing curly braces (}') in a section may accidentally completely end the WaveSchedule.</p> <p>May be useful when debugging large pop files.</p>

Warning - W0303

Message	Using When with MinInterval and MaxInterval may stop spawning midwave
Config	bool_warn_min_max_interval_stop_spawn Default: true
Example	<pre>PeriodicSpawn { Where spawnbot When // W0303 { MinInterval 1 MaxInterval 30 } TFBot { Class Pyro Skill Easy } }</pre>
Explanation	When can alternatively be used with a single constant value rather than expanding to MinInterval and MaxInterval. Using MinInterval and MaxInterval may occasionally completely break the PeriodicSpawn.

Errors

When an error criteria is met, P3 will terminate and a message will be added to the screen.

An error means that a critical issue has been found in the pop file.

All prior warnings are displayed on termination. All future warnings and errors cannot be safely determined due to the unknown severity of the error.

For example, a chain reaction of additional warnings and errors may occur due to the existence of an early error, therefore P3 immediately terminates on the detection of the first error.

Each error is associated with a unique identifier which may be used to look up its associated error criteria in this guide.

Errors are not configurable. Changes to the pop file must be made in order to resolve the error.

—

Errors may very rarely occur without termination. These types of errors should also not be taken lightly as they may still crash TF2 on attempt to load the pop file.

An example of this is the failed attempt to parse a base file, code E0801.

Error - E0501

Message	Invalid symbol found near X X - Relative location of error
Example 1	<pre>WaveSchedule { ThisCollectionIsInvalid // E0501 { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Scout Skill Easy } } } }</pre>
Explanation 1	<i>ThisCollectionIsInvalid</i> is not a valid term. A possible fix here is replacing <i>ThisCollectionIsInvalid</i> with <i>Wave</i> .
Example 2	<pre>WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot Item "The Shortstop" // E0501 TFBot { Class Scout Skill Easy } }</pre>
Explanation 2	The <i>Item</i> key-value belongs inside TFBot instead of WaveSpawn. This also occurs with any misplaced key-value.

Error - E0502

Message	Bad comment found near X X - Relative location of error
Example	<pre>// The following line does NOT produce an error Wave //Comment { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Scout Skill Easy } } } // The following line is error E0502 Wave//Comment { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Scout Skill Easy } } }</pre>
Explanation	<p>The first part of this example shows a valid comment. <i>Wave</i> becomes <i>Wave//Comment</i> in the second part of this example. This results in an error due to the lack of a space between <i>Wave</i> and the double slashes (<i>//</i>).</p> <p>This error only affects collections (terms that come right before an open curly brace).</p>

Error - E0503

Message	Cannot have nested complex spawners: 'X' X - Inner complex spawner name
Example	<pre>RandomChoice // Outer complex spawner { Squad // Inner complex spawner, E0503 { TFBot { Class Heavy } TFBot { Class Heavy } } Squad { TFBot { Class Soldier } TFBot { Class Soldier } } }</pre>
Explanation	<p>A complex spawner is a spawner that is designed to only contain simple spawners.</p> <p>Complex Spawners: Squad, Mob, RandomChoice Simple Spawners: TFBot, Tank, SentryGun</p> <p>TF2 has decided to not support nested complex spawners due to the recursive implementation of the enemy spawner.</p> <p>Although the pop file may still function, the random or group nature of the complex spawners may unreliably exist.</p> <p>The icons of the involved spawners may become invisible, and the wave progress bar may become compromised.</p> <p>This error occurs with any combination of complex spawners of any nested degree.</p>

Error - E0801

Message	Could not parse Template file 'X' at location: Y X - Base file name Y - Assumed base file path on disk No line number shown. Does not terminate.
Example 1	<pre>// Contents of badtemplate.pop TemplateSchedule { Templates// E0502 Bad Comment { AddHatTemplate { Item "Ghostly Gibus" } } } // Contents of mymission.pop #base badtemplate.pop // E0801 occurs here WaveSchedule { Wave { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Scout Template AddHatTemplate // W0211 Collateral } } } }</pre>
Explanation 1	<p>This example consists of 2 files: mymission.pop and badtemplate.pop mymission.pop uses badtemplate.pop as a #base template file.</p> <p>badtemplate.pop produces an error code E0502 (Bad Comment) when parsed independently and crashes the game when loaded.</p> <p>The error in the template results in a collateral warning issued to all attempts at accessing templates.</p>

<p>Example 2</p>	<pre>// Contents of goodtemplate.pop TemplateSchedule { Templates { AddHatTemplate { Item "Ghostly Gibus" } } } // Contents of mymission.pop #base goodtemplate // E0801 occurs here WaveSchedule { Wave { WaveSpawn { TotalCount 1 MaxActive 1 SpawnCount 1 Where spawnbot TFBot { Class Scout Template AddHatTemplate // W0211 Collateral } } } }</pre>
<p>Explanation 2</p>	<p>This example consists of 2 files: <i>mymission.pop</i> and <i>goodtemplate.pop</i>. <i>mymission.pop</i> uses <i>goodtemplate</i> as a #base template file.</p> <p>The #base value is <i>goodtemplate</i> which triggers E0801 because the <i>.pop</i> file extension is required when specifying a template file.</p> <p>The desired template file is not parsed due to being unable to locate the file.</p> <p>The error in the template results in a collateral warning issued to all attempts at accessing templates.</p>

<p>Example 3</p>	<pre>// Location of goodtemplate.pop C:/Folder/goodtemplate.pop // Location of mymission.pop C:/DifferentFolder/mymission.pop</pre>
<p>Explanation 3</p>	<p>This example consists of 2 files: <i>mymission.pop</i> and <i>goodtemplate.pop</i>. <i>mymission.pop</i> uses <i>goodtemplate.pop</i> as a #base template file. Both files are perfectly written but in different places.</p> <p>P3 assumes all template files used are in the same folder as the parsed pop file.</p> <p>Keep mission pop files in the same directory as your template pop files in order to use them.</p>

Error - E0997

Message	MESSAGE MAY VERY
Example	N/A
Explanation	<p>Something internal failed. Redownload P3 and try again.</p> <p>If problem persists, please contact the developer so he can fix it.</p> <p>Contact information can be found in the README.md and the GitHub repository.</p>

Error - E0998

Message	MESSAGE MAY VERY Does not terminate.
Example	N/A
Explanation	<p>On extremely rare instances, the parser may completely fail due to an unknown bug.</p> <p>If this error occurs, please contact the developer so he can fix it.</p> <p>Contact information can be found in the README.md and the GitHub repository.</p>

Error - E0999

Message	MESSAGE MAY VERY
Example	N/A
Explanation	<p>On extremely rare instances, the parser may completely fail due to an unknown bug.</p> <p>If this error occurs, please contact the developer so he can fix it.</p> <p>Contact information can be found in the README.md and the GitHub repository.</p>