



Novel Games

**Compiled Single Player Games
Installation Guide**

Version 1.6.3



Table of Contents

1. Introduction.....	3
2. Directory Structure.....	4
3. Step-by-Step Example.....	5
4. Parameters on the Server Side	7
5. Obscene Names Filter	9
6. Parameters to Pass to the Game	10
7. Troubleshooting	12
8. Document Modification History	14



1. Introduction

This document describes the way to install the compiled single player games (HTML5, Java JAR) into your web server.



2. Directory Structure

The files are arranged in the following directory structure:

```
+
+-- instructions.pdf
+-- LICENSE.TXT
+-- games
+-- icons
+-- versions
+-- PHPMySQL (if using Leaderboard)
```

The following are the descriptions of each of the files / folders:

- instructions.pdf
 - This file
- LICENSE.TXT
 - The license agreement
- games
 - The folder containing the game files and sample HTML pages
- icons
 - The folder containing the icons and screenshots of each of the games, you can use these files in your web page or other areas
- versions
 - The folder containing the release notes of each of the games, this is for your reference
- PHPMySQL
 - This folder exists only if you are using leaderboard. This folder contains the server side PHP files and SQL schema for you to upload to your web server and set up the database



3. Step-by-Step Example

This section walks through a step-by-step example of installing the games in your web server.

1. Upload the files and folders in the `games` folder to a directory in your web server
2. If you are not using a leaderboard, then the process is complete. You can now open a web browser and browse to the HTML files in the directory to play the game. The HTML files are meant to be sample files only, you can modify them to suit your needs.
3. If you are using a leaderboard, then there are more steps needed to set up the database and upload the server side scripts to your server. The remaining of this document are relevant only if you are using a leaderboard.

Leaderboard Set Up

1. Run the SQL statements in `PHPMySQL/table.sql` to create a table in your database to store the results.
2. Edit `$databaseHost`, `$databaseUser`, `$databasePassword` and `$databaseName` in `config.php` so that they contain the actual database host name, database user name, database password and database name respectively.
3. Upload all the files except `table.sql` under the `PHPMySQL` directory to your web server. The files should be put in the same directory you have put the games to. That is, the `games` directory and the `PHPMySQL` directory merge into one single directory in your web server. If you would like the server side scripts to be in a different directory, please refer to the "Troubleshooting" section.
4. Modify the `pictures` directory so that the web server has read-and-write access to it.
5. Navigate to `test.php` in your web browser. This script will run a few tests to see if the settings are correct. If errors occur, please check the settings again.
6. That's all of the setup. Now navigate to the game HTML files in your web browser and test the games. The results of the games should now be able to be saved and loaded.



7. You can navigate to `manage.php` to perform various management activities such as deleting records in the leaderboard. By default the username and password are both **admin**. You can modify `config.php` to change these.



4. Parameters on the Server Side

This section explains the parameters that can be tuned on the server side. These parameters are put in the config.php file.

Parameter	Description
\$databaseHost	The host of the database.
\$databaseUser	The username of the database.
\$databasePassword	The password of the database.
\$databaseName	The name of the database.
\$databasePort	The port of the database.
\$databaseTimeZone	The time zone of the database.
\$databaseTableName	The name of the table in the database containing the scores.
\$databaseGameCategoriesTableName	The name of the table in the database containing the games categories
\$databaseGameCategoriesGameIDField	The name of the game ID field in the game categories table
\$databaseGameCategoriesCategoryIDField	The name of the category ID field in the game categories table
\$databaseCoinsPowerUpsLogTableName	The name of the table in the database containing the records of coins and power ups gains and uses
\$databaseGrandMastersTableName	The name of the table in the database containing the information about grand masteries
\$databasePlayersTableName	The name of the table in the database containing the information about players
\$powerUpsPrices	The coins worth of each of the power ups
\$coinsPurchasePrices	The real prices of the coins
\$controlUsername	The username for the manage.php script.
\$controlPassword	The password for the manage.php script.
\$paypalClientID	The PayPal client ID for receiving money when the player purchase coins
\$paypalSecret	The PayPal secret
\$paypalAPIHost	The PayPal API host
\$paypalPayeeEmail	The PayPal email for receiving money
\$paypalErrorSupport	The email shown to the player if an error occurs during



tEmail	purchase
\$picturesPath	The path to the folder where the pictures will be stored.
\$hashKey	The hash key used. Do not change it if the hash key is already set.
\$hashKeyRequired	Whether a check for the hash to authenticate the submitted score is required.
\$shareTexts	The texts used when sharing for different number of stars earned.
\$shareFacebookAppID	The Facebook app ID. This is required only if login through Facebook is allowed.
\$shareFacebookAppSecret	The Facebook app secret. This is required if login through Facebook is allowed.
\$twitterConsumerKey	The twitter consumer key. This is required if login through twitter is allowed.
\$twitterConsumerSecret	The twitter consumer secret. This is required if login through twitter is allowed.
\$googleAPIClientID	The Google API client ID. This is required if login through Google is allowed.
\$googleAPIClientSecret	The Google API client secret. This is required if login through Google is allowed.
\$maintenanceUndergoing	Whether or not maintenance is undergoing. If maintenance is undergoing, all write operations to the database will be blocked.
\$maintenanceUndergoingUntilUTC	The time in UTC at which maintenance is expected to end.



5. Obscene Names Filter

An obscene names filter is installed so that obscene names will not be input to the database. You can edit `obscene.txt` to add or remove obscene words.

If an obscene word (ignoring spaces) is a part of the game entered by the user, the name will be rejected.

Alternatively you can implement `checkNames` function in `config.php` to implement your own algorithm to filter out obscene words.



6. Parameters to Pass to the Game

This section explains the parameters that can be passed to the game. These parameters should be passed from the HTML file to the game. Depending on the format of the games you licensed, the way to pass is slightly different:

Java JAR

In this case you pass the variables through applet parameters, e.g.

```
<script src="http://www.java.com/js/deployJava.js"></script>
<script>
var attributes = { code:'com.novelgames.spgames.lightning.Main',
width:600, height:400};
  var parameters = {   jnlp_href:   'lightning_applet.jnlp',
playerName: 'John Smith'};
  deployJava.runApplet(attributes, parameters, '1.7');
</script>
```

HTML5

In this case you pass the variables through JavaScript, e.g.

```
<script>
nomic.initialize(
  document.getElementById('mainCanvas'),
  { playerName: 'John Smith' }
);
</script>
```

Parameter	Description
siteID	If this parameter is used, its value will override the site ID set in the game.
gameID	If this parameter is used, its value will override the game ID set in the game.
gameName	If this parameter is used, its value will override the game name set in



	the game.
playerName	This parameter controls the name of the player. If this parameter is not present, the player will need to enter his / her name before submitting the result to the leaderboard. If this parameter is present, the result will be submitted to the leaderboard using this name.
playerPictureURL	This parameter controls the URL of the player's picture. If this parameter is present, a picture of the player will be displayed in the leaderboard.
playerFacebookUserID	This parameter sets the Facebook user ID of the playing player.
playerTwitterUserID	This parameter sets the Twitter user ID of the playing player.
playerGoogleUserID	This parameter sets the Google user ID of the playing player.



7. Troubleshooting

Q) The leaderboard is not working.

A) Please check

1. Whether you can run the `test.php` file without errors;
2. Whether all the files, PHP, JS, HTM, JAR, etc are under the same directory.

Q) We want to organize the scripts so that the server side scripts and the game files are in different folders. Is this possible?

A) Yes, you can add a “base” parameter to the game through the HTML page. The following examples show the case when the game is accessed by `http://www.yoursite.com/games/game.htm` and the server side scripts are put in `http://www.yoursite.com/php/` folder

Java JAR

```
<script src="http://www.java.com/js/deployJava.js"></script>
<script>
var attributes = { code:'com.novelgames.spgames.lightning.Main',
width:600, height:400};
var parameters = { jnlp_href: 'lightning_applet.jnlp', base: '
http://www.yoursite.com/php/' };
deployJava.runApplet(attributes, parameters, '1.7');
</script>
```

HTML5

```
<script>
nomic.initialize(
document.getElementById('mainCanvas'),
{ base: ' http://www.yoursite.com/php/' }
);
</script>
```



Questions?

Please contact us at
support@novelgames.com



8. Document Modification History

Version	Date	Description
1.0.0	2014-07-08	First Draft
1.1.0	2014-08-20	Updated HTML5 codes
1.2.0	2014-10-29	Updated server side parameters
1.3.0	2018-12-04	Updated information about leaderboard
1.3.1	2019-07-03	Fixed some errors
1.3.2	2020-01-30	Added \$databasePort server side parameter
1.4.0	2020-03-26	Added section about obscene names filter
1.5.0	2020-03-31	Added \$maintenanceUndergoing and \$maintenanceUndergoingUntilUTC parameters
1.6.0	2020-05-09	Added information about coins, power ups, and grand masters. Also added function to delete records
1.6.1	2020-06-30	Removed information about Flash which is deprecated
1.6.2	2022-02-10	Updated tables and parameters
1.6.3	2022-07-27	Removed \$googleAPIKey which is no longer needed