
THE ADLIBERUM ENGINE

Thank you for choosing Adliberum Engine. Role-play Text Engine.



At its' core ADLENGINE is about text, or data. The power comes in how that data is then presented through the use of different renderers and customization.

Tweet me anytime [@liamtwise](#) or [@adliberum](#)

If you would like to support my cause then [check out my patreon](#)

Thanks again, and enjoy! I look forward to seeing what you manage to create using adlengine!

Table of Contents

MOVEMENT.....	2
INTERACTION.....	2
CHARACTER COMMANDS.....	3
DOWNLOADING ADLENGINE.....	3
SHAPING A WORLD.....	4
SETTING ROOM IMAGES.....	4
DEFINING OBJECTS.....	5
EDITING ROOMS.....	6
EDITING EXITS.....	7
SYSTEM COMMANDS.....	7
MULTIPLAYER.....	8
CHATTING WITH OTHER PLAYERS.....	10
COMMAND / NAME VARIATIONS.....	10
CUSTOM COMMANDS.....	10
COMING SOON.....	11
BUILDING YOUR FIRST ADVENTURE.....	12
COMMAND REFERENCE:.....	16

THE ADLIBERUM ENGINE

MOVEMENT

To move around in the world you use commands such as **n** or **north**

You can only move in directions that have been laid out.

- ◆ north (or **n**)
- ◆ east (or **e**)
- ◆ south (or **s**)
- ◆ west (or **w**)
- ◆ northeast (or **ne**)
- ◆ northwest (or **nw**)
- ◆ southeast (or **se**)
- ◆ southwest (or **sw**)

Other movement commands include **up** (u), **down** (d), **in**, **out** and **back**.

You could map a custom command for a room to take the player in a specific direction for example.

- ◆ if enter wardrobe then n

This would make the player go north. You could then hide the direction north by saying n is hidden.

One step further;

- ◆ if open wardrobe then n is visible
- ◆ if close wardrobe then n is hidden

INTERACTION

THE ADLIBERUM ENGINE

You can control the environment by interacting with objects, characters and more. Ultimately this will be based on the type of adventure you are playing and what the creator has granted you.

Typical interaction commands:

- ◆ take / drop
- ◆ open / close
- ◆ lock / unlock
- ◆ break / fix

- ◆ eat / drink
- ◆ wear / remove
- ◆ equip / unequip
- ◆ push / pull
- ◆ use

CHARACTER COMMANDS

- ◆ i / inv

Lists what you are carrying.

- ◆ Wearing

Lists what you are wearing.

- ◆ equipped

Lists what you have equipped.

DOWNLOADING ADLENGINE

To download the engine. There are a few options to choose from.

- a) use our online version at <https://www.adliberum.com/engine/141> (for the latest version)

THE ADLIBERUM ENGINE

b) go to <https://liamtwose.itch.io/adlengine> and download a copy for Windows, Linux or Mac.

c) download the web files from the above link and install them on your web server e.g. `yourwebsite.com/adlengine` (you could password protect this) - these web files would also be used for you to host your game as the engine is both creator and player.

d) You can also download the engine using itch.io's cool app at <https://itch.io/app> which will automatically update the engine as I release more versions. Search for 'adlengine'.

SHAPING A WORLD

Ignore [] - delete as appropriate.

Build / delete object

- ◆ **build** rope
- ◆ **delete** door

build exit [direction]

- ◆ **build exit** n
- ◆ **build exit** w

delete exit [direction]

- ◆ **delete exit** s
- ◆ **delete exit** nw

SETTING ROOM IMAGES / ADDING SCRIPT FILES / ASSETS

- Step 1: open the game folder (where the files are) | `program files\steam\steamapps\common\The Adliberum Engine (ADLENGINE)\`
- Step 2: rename `package.nw` to `package.zip` (you will need to edit the folder options to show / allow file extensions to be edited)

THE ADLIBERUM ENGINE

- Step 3: open the zip and copy any images you plan to use into it.
- Step 4: rename package.zip back to package.nw
- Step 5: relaunch adlengine
- Step 6: use... room image is imagename (ignore the file extension) and it will change the image.

DEFINING OBJECTS

Objects are defined by assigning them properties and parameters; below is a comprehensive list. Tip: not defining something instantly makes that action not possible, rather than adding that it can't. The time you might want to add these is if for example an object starts as droppable, but then becomes cursed when someone picks it up.

- ◆ **object** can be taken
- ◆ **object** can be worn
- ◆ **object** is immovable
- ◆ **object** [can / cant] be opened
- ◆ **object** [can / cant] be closed
- ◆ **object** [can / cant] be eaten OR **object** is edible
- ◆ **object** [can / cant] be drunk
- ◆ **object** is [visible / hidden]
- ◆ **object** is broken

You can also define how an object reacts to having specific actions performed on it.

- ◆ **object if** examined do some action
- ◆ **object** examined msg some message

control commands

- ◆ **object if** taken
- ◆ **object if** dropped
- ◆ **object if** opened
- ◆ **object if** closed
- ◆ **object if** locked
- ◆ **object if** unlocked
- ◆ **object if** eaten

THE ADLIBERUM ENGINE

- ◆ **object if** drunk
- ◆ **object if** equipped
- ◆ **object if** unequipped
- ◆ **object if** worn
- ◆ **object if** removed

- ◆ **object** controls

The controls command mimics the actions taken against this object on either a specific room or exit that you specify. For example. A door (object) leading north when opened, could also open the n exit. This is the best way to make an object part and parcel with a physical exit.

- ◆ **object** requires

The requires command is things like keys, skills etc.

- ◆ **object** needs

Expendable things i.e. needs wood to be built (if crafting)

- ◆ **object** uses

Uses is things like bullets, fuel; think consumables.

- ◆ **object** targets
- ◆ **object** toggles

The targets and toggles command can be used to specify an action for when a particular object is used to toggle an option, such as visibility (i.e. light switch and light)

- ◆ **object** grants

The grants command can be used to either grant health or death depending on whether you instruct your game to factor health into it.

EDITING ROOMS

- ◆ **room name is** Some Name
- ◆ **room description is** Some Description

THE ADLIBERUM ENGINE

- ◆ **room image is** trapped
- ◆ **room is** pitch black / barely lit / dark / light / well lit / bright

EDITING EXITS

- ◆ **exit e is** open
- ◆ **exit u is** closed
- ◆ **exit n is** locked
- ◆ **exit ne is** unlocked
- ◆ **exit s is** visible
- ◆ **exit s is** hidden

SYSTEM COMMANDS

- ◆ **help or dl help**

Brings up the manual in a popup OR downloads the manual to your computer. NB: The popup may get blocked by your browser.

- ◆ **map name is**

Sets the current map name to whatever you like. required for loading different areas / maps internally.

- ◆ **dl**

Downloads the current game to a .adlengine file with the same name as the map name, which you can set with the **map name is** blah command

- ◆ **load**

i.e. load dungeon will load the game file if it exists. *If you are in a multiplayer session and you are the dm / host the level you load will sync to everyone else. Great for setting the game before play.*

- ◆ **new map**

THE ADLIBERUM ENGINE

Starts a fresh map

- ◆ **update exits**

Updates all exits, useful if you are manually setting exits as locked etc

- ◆ **list**

Provides a list of objects / rooms based on search criteria specified. use list obj to list only objects.

- ◆ **goto**

Use the goto command to instantly teleport yourself to an object or room location. i.e. goto corridor, or goto gate, or goto key

- ◆ **fs** or **fullscreen**

Enables fullscreen mode

- ◆ **fs** off

Disables fullscreen mode

- ◆ **.history**

Displays a text box with complete command history, useful for sharing bugs or if you're using the engine to roleplay, to take a copy of the history.

EXPERIMENTAL (LANGUAGE) COMMANDS

.es (enables spanish keyboard)
.en (enables english keyboard)
.us (enables english keyboard)

MULTIPLAYER

- ➔ **.ghost** or **.ghost on**

THE ADLIBERUM ENGINE

The ghost command allows you to roam around the world in observer mode but you won't be able to interact with the world in any way. Use `.ghost off` to turn it off.

→ `.join`

Use the `.join` command to join a session (if you are the first to join you will become the host or DM.)

→ `.nick` name

Use the `.nick` command to set your player name in multiplayer i.e. `.nick liam`

→ `.pass`

i.e. `.pass serp3nt`

Setting a password allows you to host a private game, either for playing with only people you know OR to be able to create a world in private or with your friends.

Anyone who wants to join will need to use the `.pass` command before they `.join` (pass must be set to the same password you've set.)

1. the password can't be blank
2. must be at least 4 characters long.
3. recommended you use a strong password

→ `.creative`

Creative mode allows players to edit the world. If you are on a multiplayer game and you are the host / dm you will grant editing abilities to all players in the session.

→ `.creative on`

Creative on ALLOWS all players to edit the world.

→ `.creative off`

Creative off DISALLOWS all active players to edit the world.

THE ADLIBERUM ENGINE

CHATTING WITH OTHER PLAYERS

There are two main ways to chat with other players; Hopefully one will suit your language / keyboard layout.

◆ **say**

e.g. say hello

◆ **#**

e.g. #hello

In time you will be able to @name say something to chat privately with another player.

COMMAND / NAME VARIATIONS

Command variations allow the player to reference an object or action in a different manner.

For example box could also be referenced as wooden box.

To setup a custom command you use ==

i.e. wooden box==box

Now if the player types wooden box OR box it will process it as one entity.

You can also use this to create variations on commands. For example open the door==open door

CUSTOM COMMANDS

Custom commands grant you to do a lot more in the engine. They can also be activated or deactivated as needed.

◆ **if** custom command **then** do something

Custom commands can call other regular commands, create or call custom commands or even scripted commands such as room is pitch black.

Take this light / lightswitch example.

1. build light
2. build lightswitch

THE ADLIBERUM ENGINE

3. `if` turn light on then room is bright
4. `if` turn light on then room is dark

we could also include deactivate / activate when switching the light on and off.

1. `build` light
2. `build` lightswitch
3. `if` turn light on `then` room is bright `and` deactivate turn light on
4. `if` turn light off `then` room is dark `and` deactivate turn light off
5. `if` room is bright `then` activate turn light off
6. `if` room is dark `then` activate turn light on

- ◆ `on` custom command `say` something

The `on say` command allows you to provide a message of feedback to the player. e.g. `on drop skull say The skull smashes to pieces`

Clears all Custom Commands

- ◆ `clear custom`

One area I will be adding will be the ability to perform simple checks when running commands, so you could do something like;

`if bucket is empty then fill bucket else say its already full` (rough example)

COMING SOON

Player control:

Commands like `if player health is 0`

Ability to set player variables, e.g.

- `player` name `is` ??
- `player` age `is` ??
- `player` health `is` full

Room based Timed / Special Events

THE ADLIBERUM ENGINE

- **every tick** do something
- **every minute** do something
- **every second** do something
- **every hour** do something

- **after 1 minute** do something
- **after 1 hour** do something

BUILDING YOUR FIRST ADVENTURE

Your world is built up of rooms, think of these rooms as separate locations connected by exits.

Now, what better way to get a feel for the engine than to create an adventure together. Ready?

First things first...

Download:

<https://liamtwose.itch.io/adlengine>

Play live:

<https://www.adliberum.com/engine/141>

Ready?

Then let's make a simple escape game.

- ◆ use the command **new map** to start a fresh adventure.

The player will start in a locked room.

Let's set up our main room.

- ◆ **room name is** Trapped
- ◆ **room description is** You find yourself alone, in a dungeon. Before you lies a stone plinth, atop, there lies a skull.
- ◆ **room image is** trapped
- ◆ **room is** dark

THE ADLIBERUM ENGINE

Let's add our two props..

- ◆ **build** plinth
- ◆ **build** skull.obj

..now customize the plinth to appear as Stone Plinth which can still be interacted with using plinth.

- ◆ plinth **name is** Stone Plinth

To escape the room we're going to need an exit..

- ◆ **build exit** u
- ◆ **build** hatch.exit

..but for now we shall hide it and lock it.

- ◆ hatch **is** hidden
- ◆ hatch **is** locked

The hatch needs to take effect on the exit leading **up**; *This means opening / closing, locking / unlocking will mirror on the direction up.*

- ◆ hatch **controls** u

To open the hatch we need a key.. we can use the **requires** command..

- ◆ hatch **requires** key

Now we're going to hide a key in the skull, so when they try to take the skull it will smash revealing the key. For this we don't need to make the key, yet.

Let us now control what happens when they take the skull.. for this escape room, first we will get it to build a key and then delete the skull (that will have smashed).

- ◆ **if** take skull **then build** key.obj **and delete** skull

..plus we need to give the player some feedback to let them know what happened.

THE ADLIBERUM ENGINE

- ◆ **on** take skull **say** The skull slips to the ground and smashes, revealing a small key

Now let's create the escape route.

- ◆ **build room** u

Now let's travel to this new room..

- ◆ *up* (or *u*)

... and make it our win condition.

- ◆ **room name is** Stairs Out
- ◆ **room description is** You scramble through the hatch and rush up some stairs ahead to your freedom. Congratulations.
- ◆ **room image is** escaped

Build an exit back down..

- ◆ **build exit** d

.. and travel back down.

- ◆ *down* (or *d*)

Finally we need to lock the hatch.. but also lock the direction n..

- ◆ **exit n is** locked
- ◆ **hatch is** locked
- ◆ **hatch is** closed

.. and make it so the player can discover the hatch above.

- ◆ **if** look up **then** hatch is visible
- ◆ **if** look around **then** hatch is visible
- ◆ **if** search **then** hatch is visible
- ◆ **on** look up **say** Gazing up you notice a small hatch, with a lock on it
- ◆ **on** look around **say** You spot a small hatch, it appears locked
- ◆ **on** search **say** Bingo, you spy a small locked hatch above

- ◆ **if** open hatch **then** exit u **is** visible

THE ADLIBERUM ENGINE

To win the game, the player needs to do the following;

- ◆ **take** skull
- ◆ **take** key
- ◆ look u (reveal hatch)
- ◆ **unlock** hatch
- ◆ u

All done. Now test it to make sure it works.

Finally type **dl** and press the enter key.

This will save your new scenario to a game file with the extension `.adlengine`; these files when placed in the main folder either on the PC or on the web will load when you run `adlengine`. You can distribute these along with the main files to allow others to play it.

What the game.adlengine file holds;

- ◆ Config / customization related data
- ◆ The stuff that makes the game work
- ◆ Your players' inventory
- ◆ Player specific data
- ◆ Custom commands you have set up.
- ◆ `verbASE` lookup commands (advanced)
- ◆ `verbOSE` lookup commands (advanced)

To have more than one map, for example if you have a map transition, you can simply call the command `load mapname`

Also in each map, if you type map name is `mapname` when downloading it will download to `mapname.adlengine`, allowing you to have separate maps / areas.

TIP: Notice the `.obj` / `.exit` on the end of some of the commands above?

Adliberum engine will eventually come complete with a whole host of standard prefabs. These classes define what properties the objects you build start off with.

For example;

THE ADLIBERUM ENGINE

- ◆ .obj will create a basic object that can be taken.
- ◆ .food will create a basic object that can be eaten.
- ◆ .drink will create a basic object that can be drunk.
- ◆ .exit will create a door complete with lock, etc.
- ◆ .chest / .container / .box will create an object that can hold other objects and can be locked / opened.

COMMAND REFERENCE:

COMMAND	DESCRIPTION	USAGE
SYSTEM COMMANDS		
help dl help	loads the manual downloads the manual	help dl help
@liamtwise	launches my twitter profile	@liamtwise
@adliberum	launches adlengines twitter profile	@adliberum
.creative	turns creative mode on	.creative / .creative [on/off]
speech on speech off	Turns speech synthesis on / off	speech on speech off
list voices voice is ..	lists the voices available set the current voice	list voices voice is 3
pause play p	pauses voice synthesis plays voice synthesis toggles voice play / pause	pause play p
list	lists anything with search parameters specified, i.e. objects, rooms etc.	list lists everything. list door would list all objects / rooms with door in the name.
goto	teleports the player to an	goto steel door

THE ADLIBERUM ENGINE

	object or room.	goto dungeon
load	loads a .adlengine map file	load .. e.g. load dungeon ignore .adlengine extension.
dl	Downloads the map data to a .adlengine file, which can be shared with others to play. SET MAP NAME IS to set the name of the file when downloaded. (it will download as game.adlengine as default)	dl
Import global import run	imports a .verbose script file into a custom command the same as the filename. .verbose files are collections of commands that are run in order. e.g. build box.verbose will be stored in custom command build box. putting global in front means it will run anywhere, when the custom command is typed / called. PLEASE SEE INSTRUCTIONS FOR WHERE TO PUT SCRIPTS!	import build box run build box
LANGUAGE COMMANDS (EXPERIMENTAL)		
.uk .en .fr .it .de .es .ru		

THE ADLIBERUM ENGINE

MOVEMENT		
u d	go up go down	u / up d / down
n e s w	go north go east go south go west	n / north e / east s / south w / west
ne nw se sw	go north east go north west go south east go south west	ne / northeast / north east nw / northwest / north west se / southeast / south east sw / southwest / south west
INTERACTION		
look	clears the screen and looks at the room you are in.	look
look at	looks at an object in detail (reading the description), if the object has a picture in the package file of the same name it will display it as a popup.	Look at box
take drop	take an object drop an object	take box drop box
open close	open an object close an object	open door close door
lock unlock	lock an object unlock an object	lock chest unlock chest
push pull	push an object pull an object	push lever pull lever
wear remove	wear an object remove an object	wear clothes remove clothes
eat drink	eat an object drink an object	eat apple drink water
equip unequip	equip an object unequip an object	equip sword unequip sword

THE ADLIBERUM ENGINE

use read	uses an object reads an object	use control panel read old book
i worn eq	Lists what you are carrying lists what you are wearing lists what you have equipped.	i worn eq
BUILDING / EDITING EXITS		
build exit .. delete exit .. exit .. is ..	builds a one-way exit in the direction you specify. deletes the exit in the direction you specify. allows you to set exits up within the rooms.	build exit ne delete exit s exit ne is [active/inactive] exit .. is hidden exit .. is visible exit .. is open exit .. is closed exit .. is locked exit .. is unlocked
EDITING THE MAP		
new map	creates a new empty map	new map
map name is	Set the name of the mapname (setting this will download the data to the same name i.e. dungeon.adlengine)	map name is dungeon
EDITING ROOMS		
room name is	set the name of the current room	room name is dungeon
room description is	sets the description of the current room	Room description is You are standing in a dank, dingy little dungeon, with only a

THE ADLIBERUM ENGINE

		skull for company
ADDING / EDITING OBJECTS		
Build	build an object objects can also inherit properties by specifying a class with the build.	build key build key.obj (creates a takeable object) build key.chest (creates a container with a lock)
delete	delete object / custom command	delete dirt
requires	Used for controlling what is needed to perform a specific action.. typically unlocking / locking	chest requires silver key
is visible is hidden	sets an object to visible sets an object as hidden	trip wire is visible trip switch is hidden
is openable is lockable	allows an object to be opened / closed allows an object to be locked / unlocked	door [is/is not] openable door [is/is not] lockable
is open is closed	makes an object open makes an object closed	chest is open chest is closed
is locked is unlocked	makes an object locked makes an object unlocked	chest is locked chest is unlocked
is takeable is immovable	allows an object to be taken prevents an object from being taken / moved	key is takeable stone block is immovable
is eatable is drinkable	allows an object to be eaten allows an object to be drunk	apple [is/is not] eatable potion [is/is not] drinkable
is wearable is equipable is removable	allows an object to be worn allows an object to be equipped allows an object to be removed	clothes [is/is not] wearable knife [is/is not] equipable cloak [is/is not] removable
is useable is readable	allows an object to be used allows and object to be read	control panel is useable old book is readable
to take	sets if an object is	to take dirt requires bucket

THE ADLIBERUM ENGINE

to drop to use to push to pull to eat to drink to wear to remove to equip to unequip to read	required to perform an action on the specified object.	to drop dirt requires bucket full of dirt
	""	to use well requires well handle
.. opens closes locks unlocks controls creates needs uses contains activates grants targets toggles name is description is ..	controls another object or direction. sets the name / description of an object.	lever controls steel door (all actions are mirrored) lever unlocks steel door box name is wooden box box description is Just a plain old wooden box, nothing special
global if if	sets a local condition for performing certain actions on an object. (placing global in front will perform it anywhere in the map).	global if drink water room is dark .. if taken if dropped if eaten if drunk if read if worn if equipped if removed if pushed if pulled if opened if closed if locked if unlocked if built if destroyed if used if examined ..

THE ADLIBERUM ENGINE

		if .. then .. [and] [..] e.g. if look up then hatch is visible and exit u is visible
global on on	provide feedback for specific actions.	on .. say .. e.g. on look up say you notice a small hatch above you
TIMED COMMANDS		
after	sets an event / action to occur after so long...	after .. second[s] .. after .. minute[s] .. after .. hour[s] .. after .. day[s] .. after .. week[s] .. e.g. door==steel door steel door if opened then after 5 seconds steel door is closed and after 6 seconds steel door is locked.
every	sets an event / action to occur every so often...	every second .. every minute .. every hour .. every day .. every week ..
MULTIPLAYER		
.join	launches a multiplayer session (if you are the first to launch, you will be made the Host / DM)	.join OR .join [pass]
.ghost	join the session as a spectator (you can't interact with the world or edit it in ghost mode)	.ghost [on/off]
.server	sets the current server name	.server dummy
.pass	sets a password required to be able to join	.pass word

THE ADLIBERUM ENGINE

--	--	--