

Interactive Fiction

Language Arts



Interactive Fiction

Interactive Fiction is a computer game that allows players to control the game's main character through a series of text commands. One of the most well-known text adventure games is Zork, Zork II and Zork III.

Our game will not be as vast as the Zorks, but it will still be fun!



Step #1

Play the Zork game to get a good feel for how an interactive fiction game works.

- Try it:

<https://www.pcjs.org/software/pcx86/game/infocom/zork1/>

```
Color Display
West of House
Score: 0
Moves: 0

ZORK I: The Great Underground Empire
Copyright (c) 1981, 1982, 1983 Infocom, Inc. All rights reserved.
ZORK is a registered trademark of Infocom, Inc.
Revision 88 / Serial number 840726

West of House
You are standing in an open field west of a white house, with a boarded
front door.
There is a small mailbox here.

>
```



Step #2

Now that you have a feel for the game, let's review story elements:

- Setting
- Plot
- Point of View
- Characters
- Theme
- Conflict

Story Elements

Writing a story involves knowing its important elements and organizing them in a way that is interesting and sensible to the readers. Use the prompts below to plan and organize your own story on the next page.

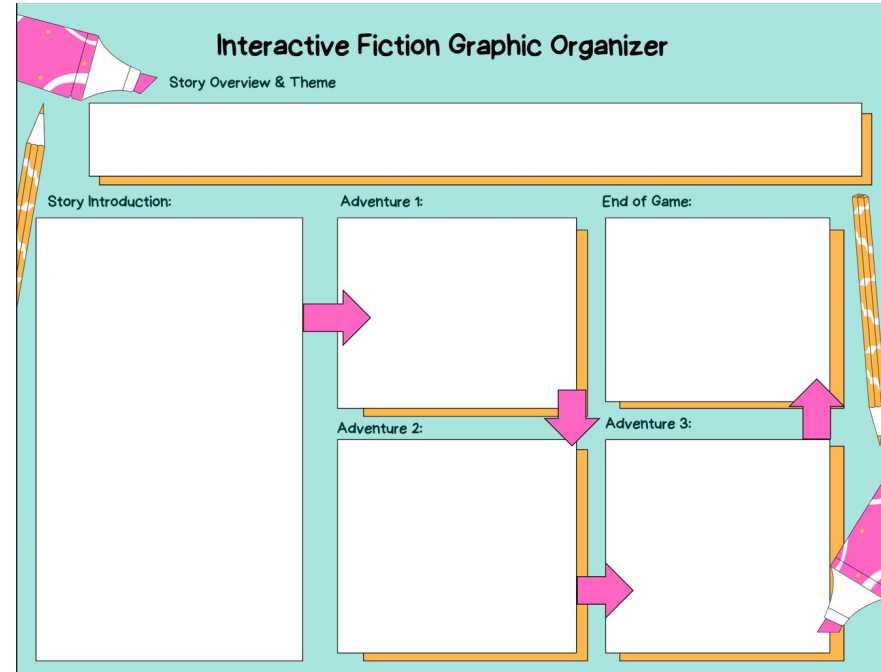
SETTING  When and where does the story happen? How does the location and time shape the plot development?	PLOT  What happens in the story? How are the events related from one another? What is the beginning, middle and ending part of the story?	CHARACTER  Who are involved and how are they portrayed in the story? How do the characters grow and change in the story?
POINT OF VIEW  Who is telling the story? What other point of view could this story be told with?	THEME  What is the message that you want to convey to your readers? What is the lesson or idea in the story that is similar to real life?	CONFLICT  What is the problem faced by the characters in the story? How is the conflict revealed in the story?



Step #3

Use a graphic organizer to draft the storyline of your game.

- Overview: What will your game be about?
- Theme: is a lesson or message explored throughout a story.
- Adventure 1: first room or conflict
- Adventure 2: second room or conflict
- Adventure 3: third room or conflict
- End of Game: how to win?



The graphic organizer is titled "Interactive Fiction Graphic Organizer" and is set against a light teal background. It features a pink highlighter and a yellow pencil on the left side. The layout includes a large white box at the top for "Story Overview & Theme". Below this, the story is organized into four main sections: "Story Introduction:", "Adventure 1:", "Adventure 2:", and "End of Game:". The "Adventure 1:" and "Adventure 2:" sections are arranged vertically, with a pink arrow pointing from Adventure 1 to Adventure 2. The "Adventure 2:" and "Adventure 3:" sections are arranged horizontally, with a pink arrow pointing from Adventure 2 to Adventure 3. A pink arrow also points from the "Adventure 1:" section down to the "Adventure 2:" section. The "End of Game:" section is positioned to the right of the "Adventure 1:" section. The entire organizer is framed by a light teal border.



Step #4

Now that you have a draft, let's begin coding the game. Start with:

- Imports
- Menu

```
1 from codex import *
2 import random
3 from time import sleep
4
5 #Define Menu
6 def menu():
7     display.clear()
8     display.print("~~ Menu ~~")
9     display.print("Press A- Begin Game")
10    display.print("Press B- End Game")
11    display.print()
12    while True:
13        if buttons.was_pressed(BTN_A):
14            storyIntro()
15        if buttons.was_pressed(BTN_B):
16            endGame()
17            break
18
```



Step #5

Next, define variables and create 2 lists of actions:

- aList: Fight actions
- bList: Hide actions

```
#Define variables for StoryIntro  
aList = ["poke", "fight", "battle", "throw rocks", "bully monster"]  
bList = ["hide", "crawl up in fetal position", "run away", "create distraction"]
```



Step #6

We will use many functions in this program. This first function is the story intro to display to screen

```
#Beginning of story frame to print to screen
def storyIntro():
    display.print("Welcome to Sarah's House of Horrors")
    sleep(1)
    display.print()
    display.print("You will have to be very clever to escape")
    sleep(1)
    display.print("Watch out for the villainous monster...")
    sleep(2)
    display.clear()
    display.print("Choose your first action to defeat the monster: ")
    display.print("Press L- to Fight; Press R- to Hide")
```



Step #7

Inside the storyIntro function, under the print statements, add a while True loop:

- If button L was pressed a random action from list a will be chosen
- If button R was pressed a random action from list a will be chosen

```
while True:
    if buttons.was_pressed(BTN_L):
        display.clear()
        num = random.randrange(4)
        action1 = aList[num]
        display.print("You have chosen to Fight!")
        sleep(1)
        display.print("You " + action1 + " at the monster and have a direct hit!")
        sleep(1)
        display.print("Now. Run!")
        storyPartTwo()
    if buttons.was_pressed(BTN_R):
        display.clear()
        num = random.randrange(4)
        action1 = bList[num]
        display.print("You have chosen to HIDE!")
        sleep(1)
        display.print("You " + action1 + " from the monster and he exits the room.")
        sleep(1)
        display.print("Now. Run!")
        storyPartTwo()
```



Step #8 -

Just like the storyIntro, create 2 lists with actions to either fight or hide.

```
#Define variables for StoryPart2  
cList = ["play dead", "sleep", "sing", "tickle"]  
dList = ["fight", "insult monster", "text friend", "sword fight"]
```



Step #9

You can copy and paste the code from storyIntro and then edit to make the storyPartTwo function.

```
#Story Part 2
def storyPartTwo():
    display.clear()
    display.print("You escaped the first room. Continue on to Room 2...")
    sleep(2)
    display.print()
    display.print("As you are walking through the door,")
    display.print("the monster attacks!")
    sleep(2)
    display.print
    display.print("Will your FIGHT or HIDE this time?")
    sleep(2)
    display.clear()
    display.print("Choose your next action to defeat the monster:")
    display.print("Press L- to Fight; Press R- to Hide")
```



Step #10

You can copy and paste the while True loop from storyIntro function and then edit the print statements and the outcomes for each button.

```
while True:
    if buttons.was_pressed(BTN_L):
        display.clear()
        num = random.randrange(4)
        action2 = dList[num]
        display.print("You have chosen to Fight!")
        sleep(2)
        display.print("You " + action2 + " at the monster and have a direct hit!")
        sleep(2)
        display.print("Now. Run!")
        winGame()
    if buttons.was_pressed(BTN_R):
        display.clear()
        num = random.randrange(4)
        action2 = cList[num]
        display.print("You have chosen to HIDE!")
        sleep(2)
        display.print("You " + action2 + " from the monster and he finds you and ATTACKS")
        sleep(2)
        endGame()
```



Step #11

Define 2 new functions:

- winGame
- endGame

Add your call to the menu function. This will run first in the program.

```
#win Game
def winGame():
    display.clear()
    display.print("You DEFEATED the monster!")
    sleep(2)
    display.print()
    display.print("You WIN!!!!", scale=3, color=BLUE)
    while True:
        break

#End of Game
def endGame():
    display.clear()
    display.print("You chose poorly.")
    display.print()
    display.print("GAME OVER.", scale=3, color=BLUE)
    while True:
        break

#Main Program
menu()
```



Step #12 - Optional Extensions

- Add as many adventure rooms to your story as you like.
- Change your list options based on your preferences.
- Add scoring for each room.
- Add different characters.

