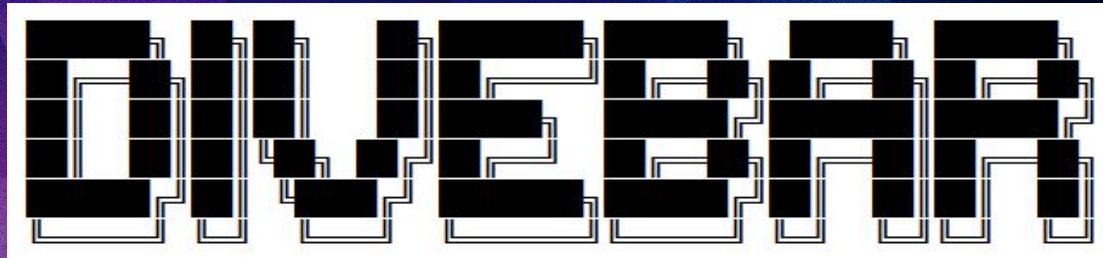
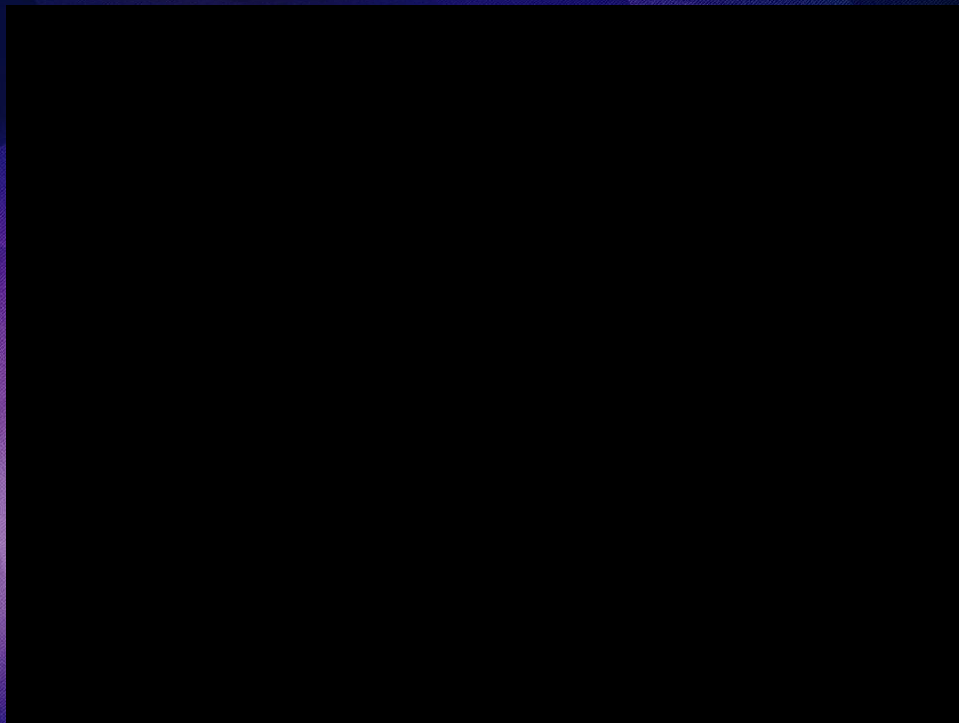


Welcome to the



What is DiveBar?



DiveBar Core Features: Customizable Experience

Two different install methods

4. You have two choices in how check for and install any Ruby Gems needed for the game

a. Run the bash script by using the command below:

```
./INSTALL.sh
```

b. **Manual installation using bundler**

- Install bundler if you don't have it already

```
install bundler
```

- Use bundler to check the gemfile and install any dependencies required

```
bundle install
```

If you run into any issues, ensure you are in the src folder. If you are still running into both problems with the script and bundler programs, you can install the gems manually by checking the gemfile to see required gems.

DiveBar Core Features: Customizable Experience

Different ways to launch the game from the command line:

```
ruby init.rb --help
#displays the help screen in the terminal. This contains rules and a command line guide

ruby init.rb -nc
#runs the program in black and white

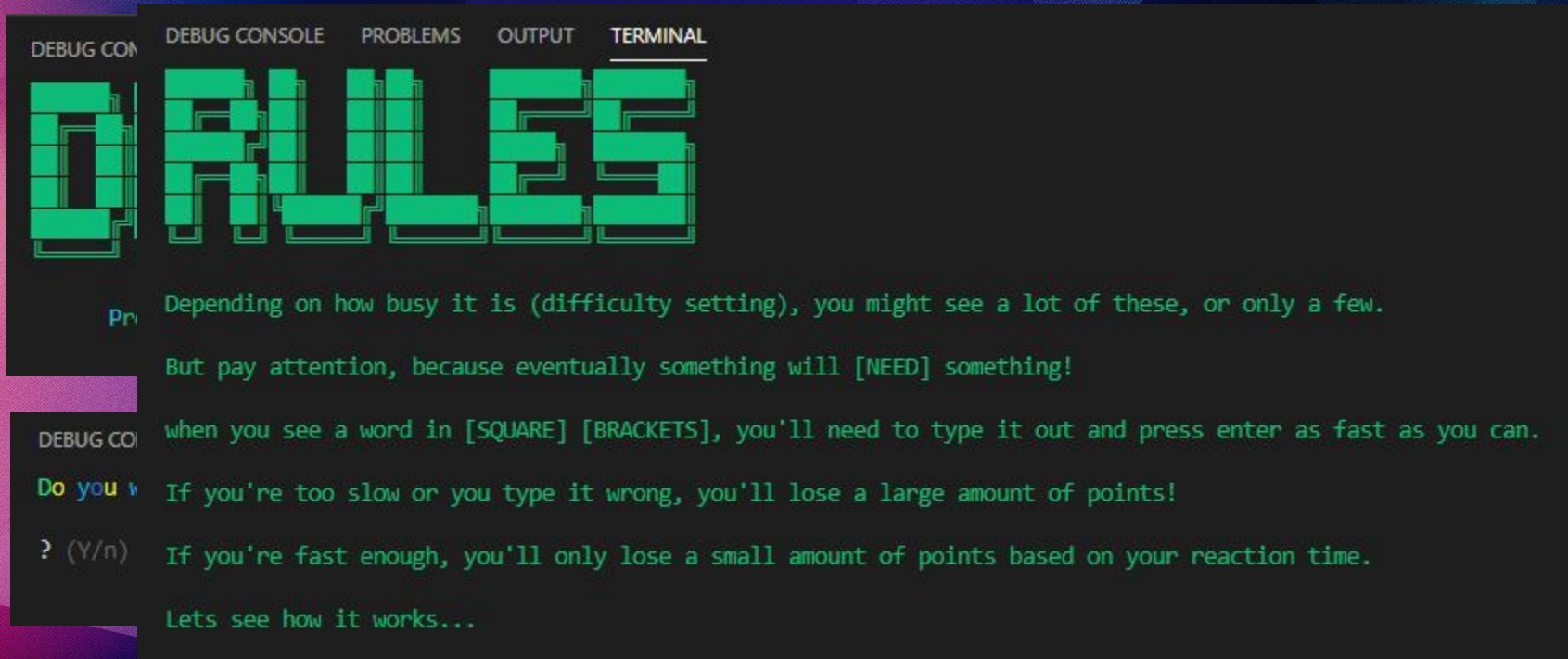
ruby init.rb -d1
#sets the difficulty of the game from the command line and skips the introduction, rules and difficulty selection screen
-d1 = easy
-d2 = medium
-d3 = hard
-d4 = insane

ruby init.rb -crash
#will crash the program, if you're into that kind of thing

You can pass the -nc argument with a -d(x) argument
If than one -d(x) argument is passed, the game will select the first one
If you pass more than two arguments, the terminal will raise an error.
```


DiveBar Core Features: Interactive Rules and Title

Responsive to user input



DiveBar Core Features: Difficulty Settings

Preset difficulty settings:

Select your desired difficulty

? (Use ↑/↓ arrow keys, press Enter to select)

- Easy
- Medium
- Hard
- Insane

```
# This module holds all the difficulty variables for the game class instance
module DifficultyModule
  # [0] difficulty symbol
  # [1] seconds given to type answer
  # [2] score penalty if timeout or wrong answer
  # [3] amount of phrases that will be displayed
  # [4] array of random sleep ranges to be chosen from each time a character is typed with the main_typer method
  # [5] number of times the phrases/prompt are flashed
  # [6] time between flashes
  # [7] string name of the difficulty
  DIFFICULTY = {
    d1: [
      :d1,
      3,
      1000,
      rand(4..6),
      [rand(0.005..0.009), rand(0.03..0.06), rand(0.03..0.06)],
      rand(3..5),
      rand(0.4..0.6),
      'EASY'
    ],
    d2: [
      :d2,
      2.5,
      2000,
      rand(3..6),
      [rand(0.005..0.009), rand(0.005..0.009), rand(0.03..0.06)],
      rand(4..6),
      rand(0.35..0.55),
      'MEDIUM'
    ]
  }
end
```


DiveBar Core Features: Randomized Output

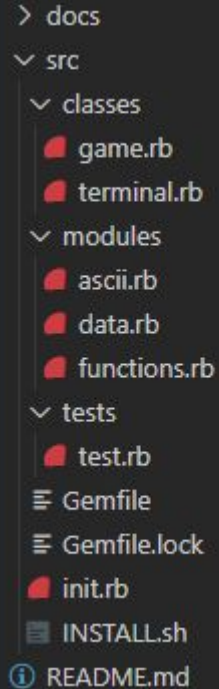
Each round is different.

```
PROMPTS = {
d1: [
    'Oi! Go and empty the [BIN] over there! NOW!',
    'You need to [PUT] those boxes away, STAT!',
    "Far out, the customers are complaining it's too [HOT] , turn on the AC!",
    "HEY! Get the swatter and kill that [FLY] , it's annoying everyone!",
    'THROW OUT THAT [OLD] WINE!',
    'You want to keep your [JOB] ? MOVE IT!',
    'Would you recommend the [RED] or the white?',
    'Oi, what can I [BUY] in this joint CHAMP?!',
    'Excuse me could you put my [BAG] behind the bar please?',
    'WE NEED MORE [ICE] RIGHT NOW!'
],
d2: [
    "Man, I've been here for 9 hours, what a [GRIND] !",
    "Who tipped the potplant over? There's [SOIL] everywhere!",
    "This is, without a [DOUBT] , the worst bar I've ever been to",
    "[SCAN] the room for glasses would you? We're running low",
    'Oh we heard this bar is all the latest [RAGE] recently!',
    'Could you [TELL] me more about where this wine is from?',
    'Mate, could I grab a [BEER] ?',
    'Just a glass of red [WINE] please',
    'Could you get a [SPARE] carton from downstairs?',
    'The amount of cockroaches in this bar is not [IDEAL]'
]
```


DiveBar Core Features: Gameplay Demo

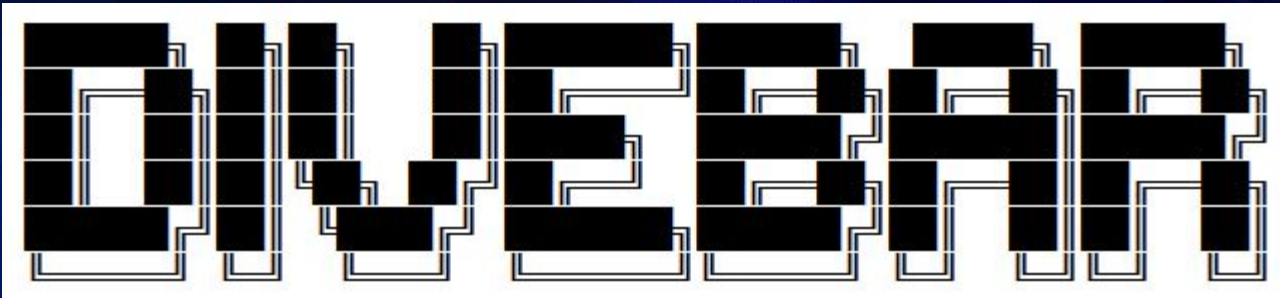
- Rules demo
-
- Confusing display tied to difficulty settings
-
- Timed input that will time out
-
- Dynamic scoring tied to your reaction time and difficulty
-
- Two different game over screens
-
- Revisit of previous features.

DiveBar Logic



```
> docs
└─ src
   └─ classes
      ├── game.rb
      └─ terminal.rb
   └─ modules
      ├── ascii.rb
      ├── data.rb
      └─ functions.rb
└─ tests
   └─ test.rb
Gemfile
Gemfile.lock
init.rb
INSTALL.sh
README.md
```

- Functions Module
- Data Module
- Ascii Module
- Terminal Class
- Game Class



Thanks for listening!

Download or fork the repo from [nick-ducker](#) on GitHub!

Have fun!