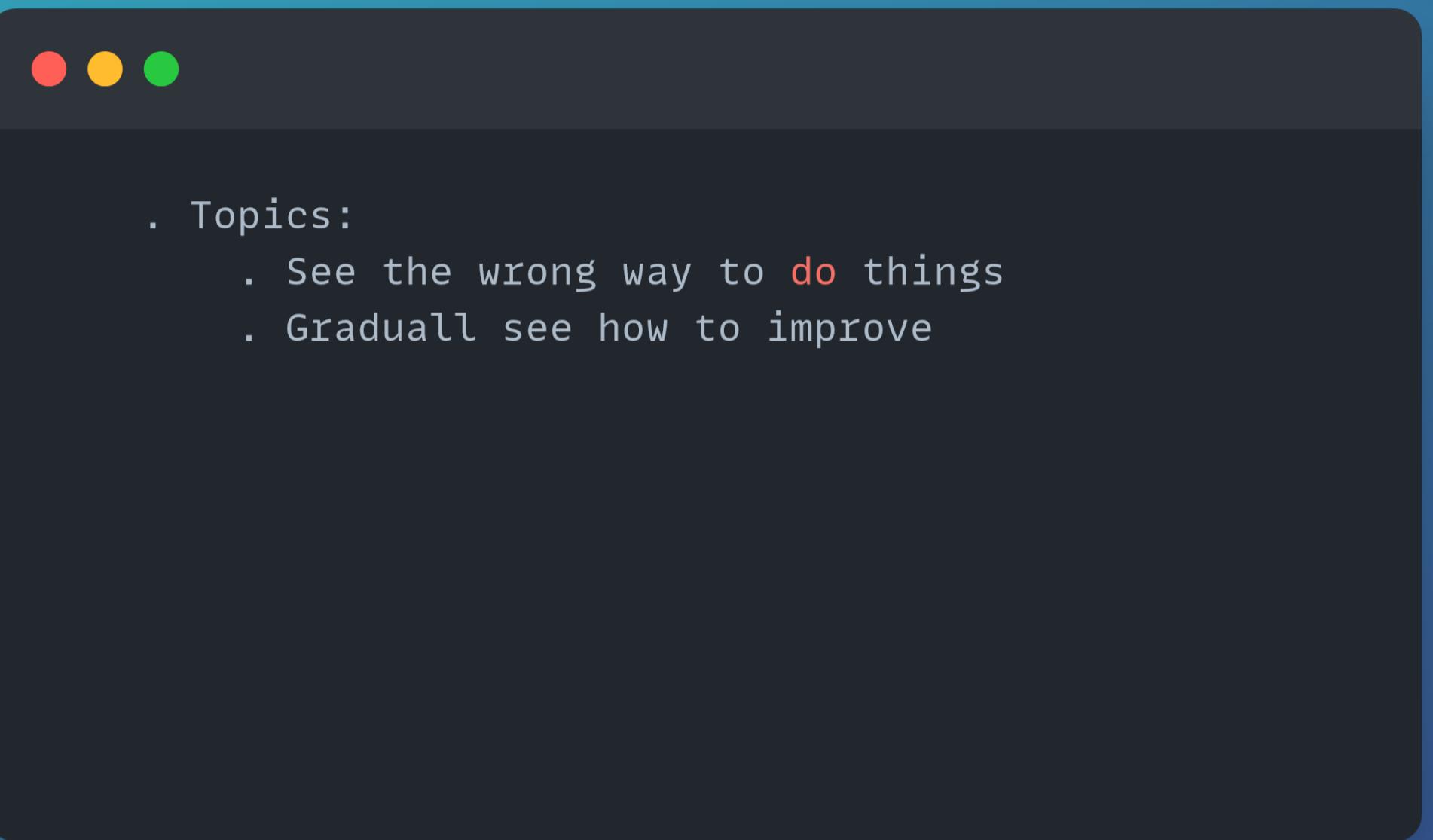


Fortune teller



Fortune teller

```
What's your name dear: Daniel
Oh dear Daniel, I see , huum, I don't see anything
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see a lot of kinds running in the backyard!
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see a lot of empty beer bottles on your work table.
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see you laughing your lungs out. I've never seen this before.
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see clouds gathering in the sky and an army standing ready for war
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see , huum, I don't see anything
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see , huum, I don't see anything
Do you want me to try again? (Y | N): y
Oh dear Daniel, I see dogs running around in a deserted empty city
Do you want me to try again? (Y | N): n
That's all I have for you today dear. Best wishes.
```

Fortune teller

```
//Store the predictions, set up random index generation and collect user name
std::vector<std::string> predictions{
    "a lot of kinds running in the backyard!",
    "a lot of empty beer bottles on your work table.",
    "you partying too much with kids wearing weird clothes.",
    "you running away from something really scary",
    "clouds gathering in the sky and an army standing ready for war",
    "dogs running around in a deserted empty city",
    "a lot of cars stuck in a terrible traffic jam",
    "you sitting in the dark typing lots of lines of code on your dirty computer",
    "you yelling at your boss. And oh no! You get fired!",
    "you laughing your lungs out. I've never seen this before.",
    ", huum, I don't see anything",
    "you drinking wine",
    "you, running away from a bear"
};

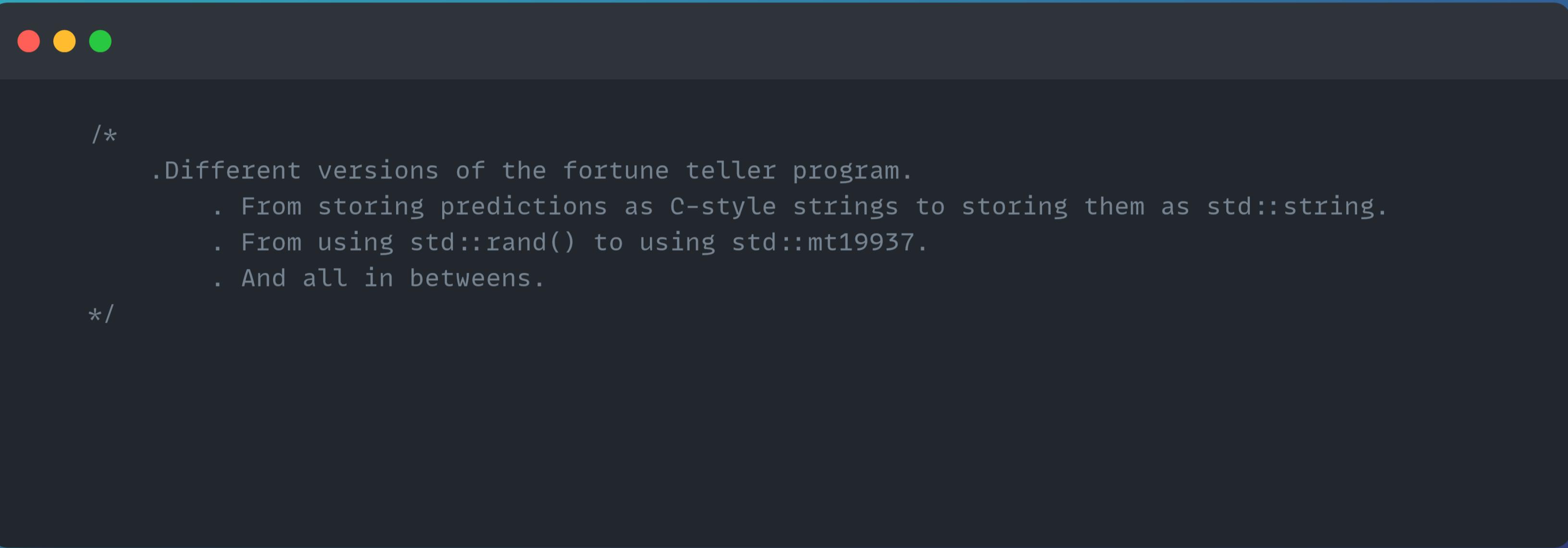
std::random_device rd;
std::mt19937 gen(rd());
std::uniform_int_distribution<size_t> dist(0, predictions.size() - 1);

bool end = false;
std::string name;

fmt::print("What's your name dear: ");
std::getline(std::cin, name);
```

Fortune teller

```
//Loop around printing random predictions until the user decides to stop by tying something  
//different from y or Y.  
while (!end) {  
    fmt::print("Oh dear {}, I see ", name);  
  
    size_t rand_num = dist(gen);  
    fmt::println("{}", predictions[rand_num]);  
  
    fmt::print("Do you want me to try again? (Y | N): ");  
  
    char go_on;  
    std::cin >> go_on;  
    std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n'); // Clear the input buffer  
  
    end = !(go_on == 'Y' || go_on == 'y');  
}  
  
fmt::println("That's all I have for you today dear. Best wishes.");
```



```
/*
.Different versions of the fortune teller program.
. From storing predictions as C-style strings to storing them as std::string.
. From using std::rand() to using std::mt19937.
. And all in betweens.
*/
```