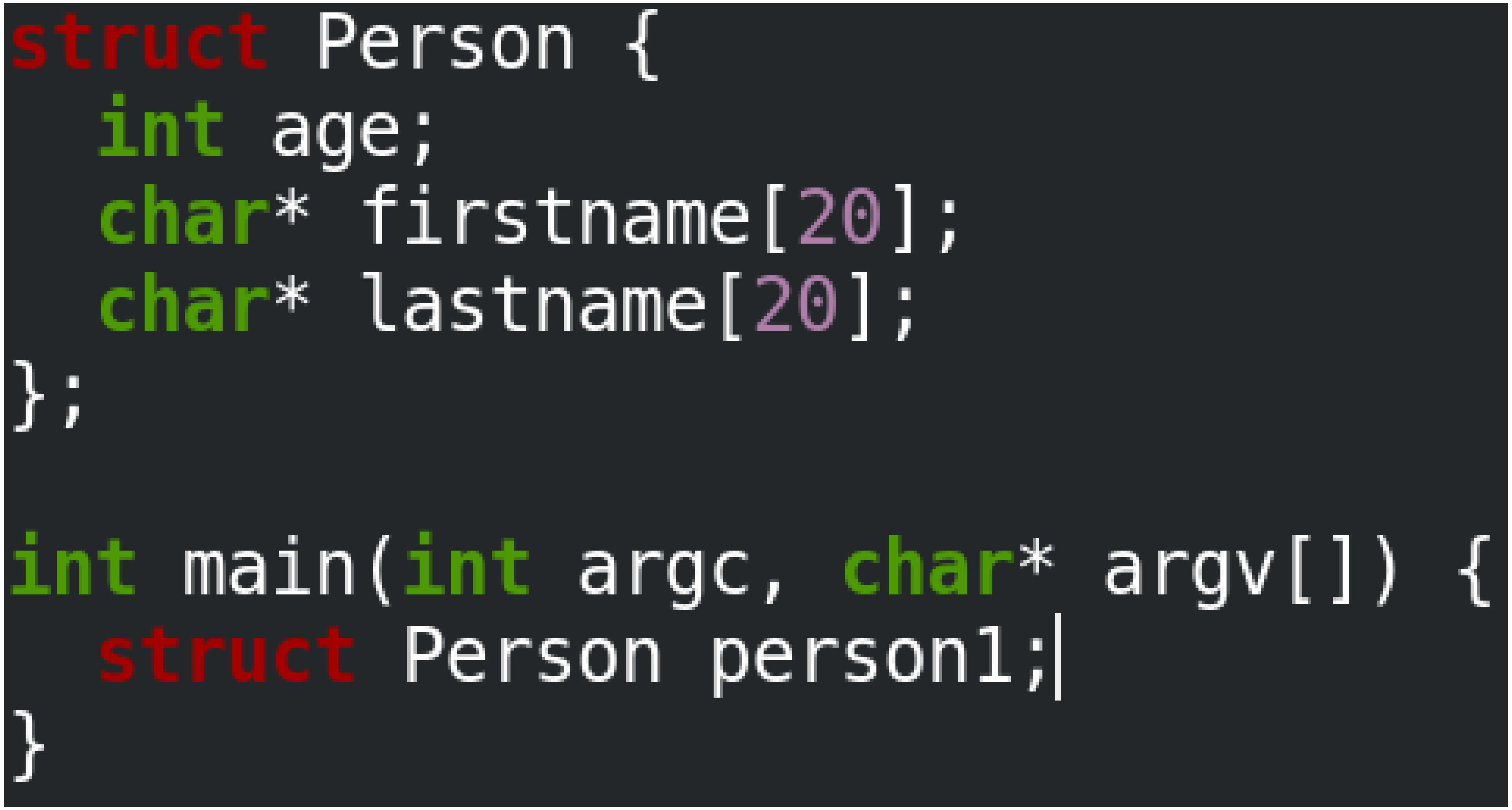
5. Using Structs in Rust

Outline

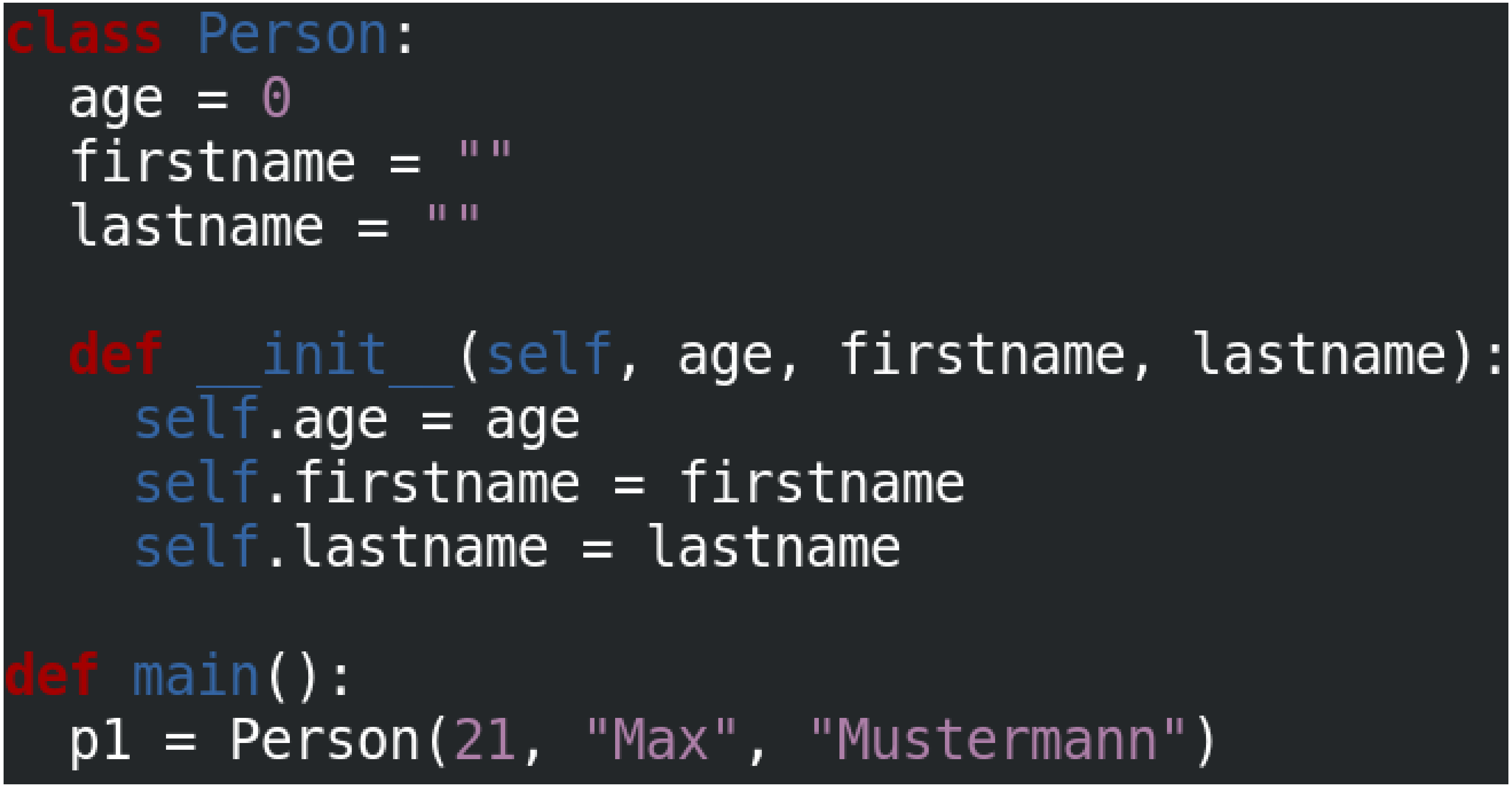
1. Structs in C / Python / Rust
2. Defining Structs
3. Instantiating Structs
4. Tuple Structs
5. The impl block

Structs in C



```c
struct Person {
  int age;
  char* firstname[20];
  char* lastname[20];
};

int main(int argc, char* argv[]) {
  struct Person person1;
}
```

Classes in Python

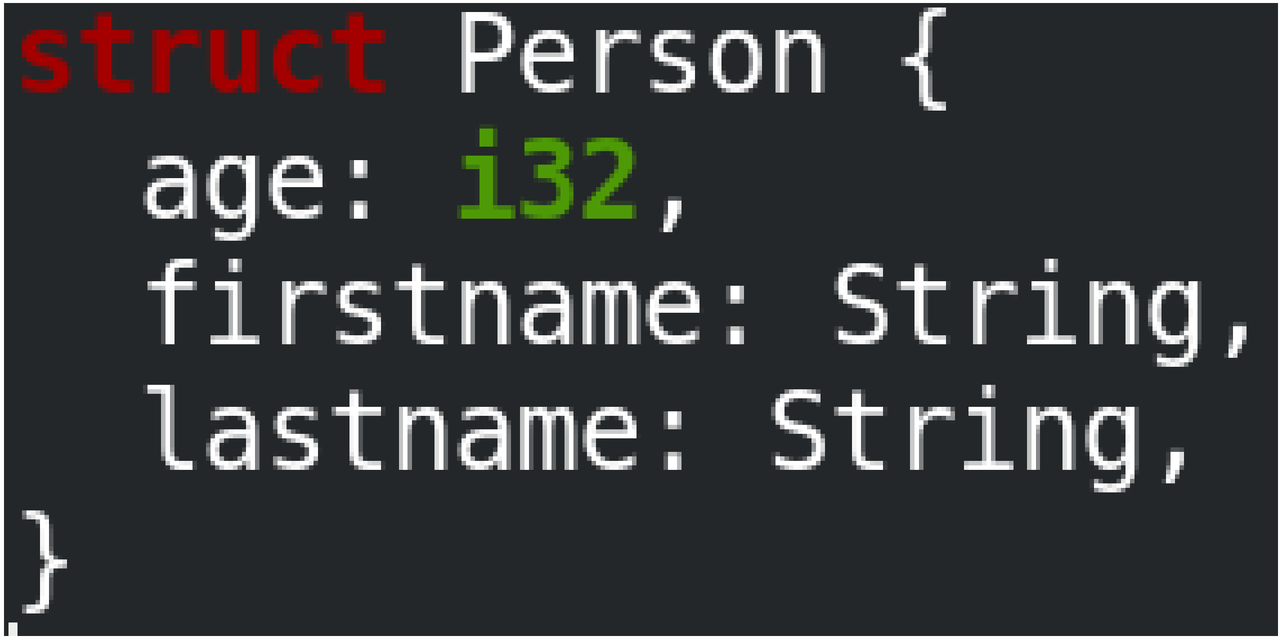


```python
class Person:
  age = 0
  firstname = ""
  lastname = ""

  def __init__(self, age, firstname, lastname):
    self.age = age
    self.firstname = firstname
    self.lastname = lastname

def main():
  p1 = Person(21, "Max", "Mustermann")
```

Structs in Rust

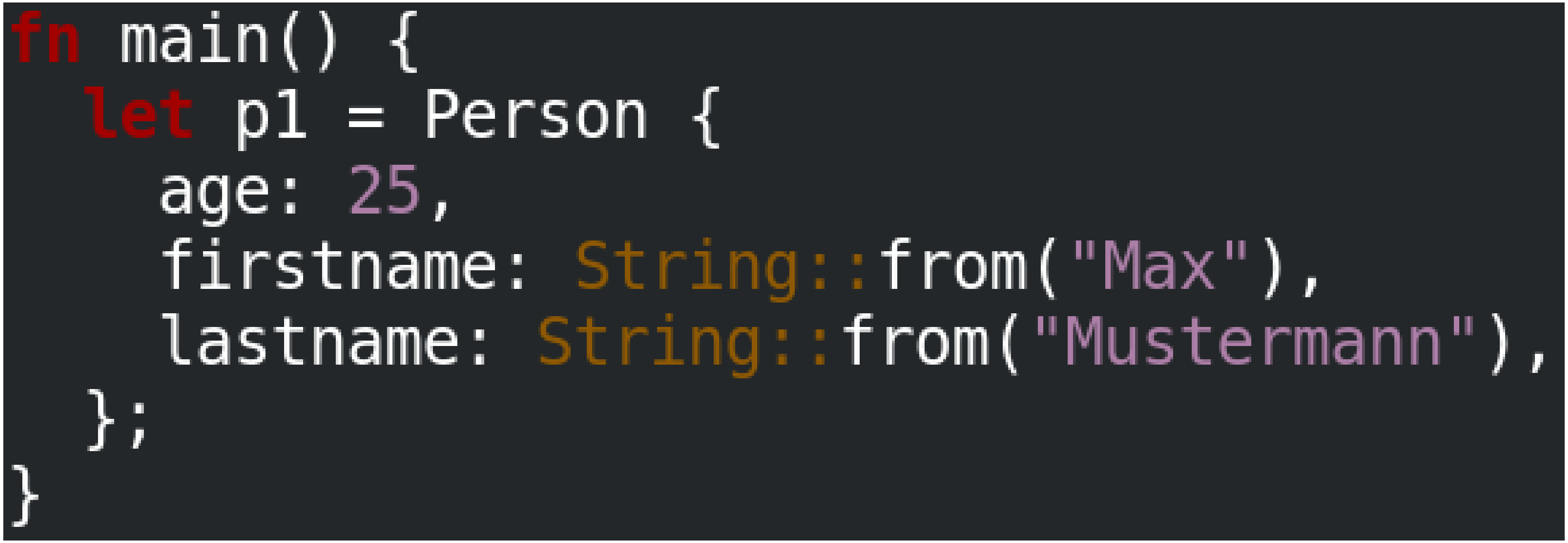
- Something between C structs and object-oriented classes
- Can be used as classes
- Consist of data and methods

Defining Structs



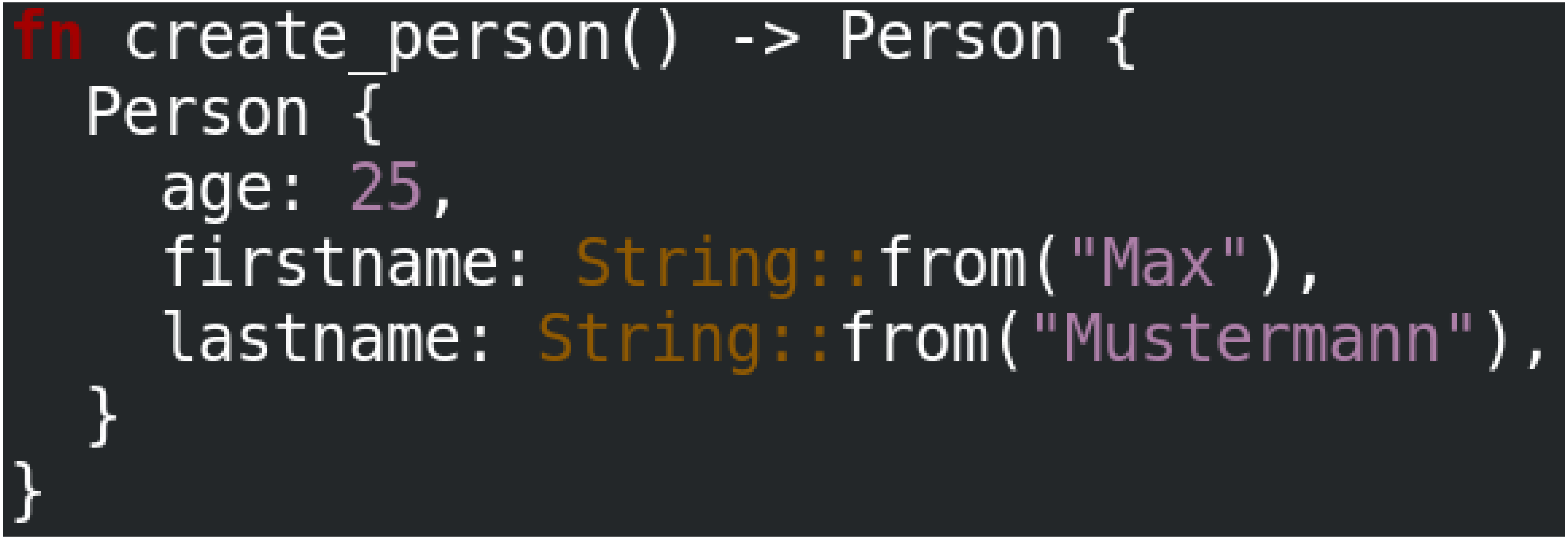
```
struct Person {
  age: i32,
  firstname: String,
  lastname: String,
}
```

Instantiating Structs (1)



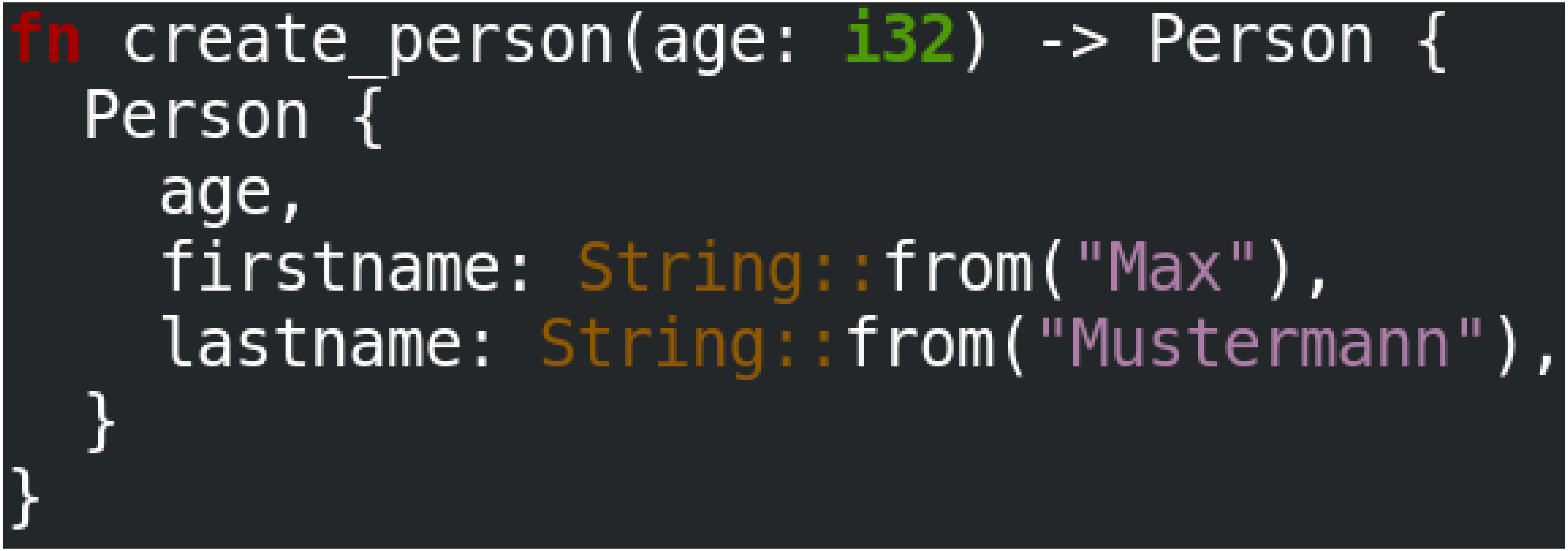
```
fn main() {
  let p1 = Person {
    age: 25,
    firstname: String::from("Max"),
    lastname: String::from("Mustermann"),
  };
}
```

Instantiating Structs (2)



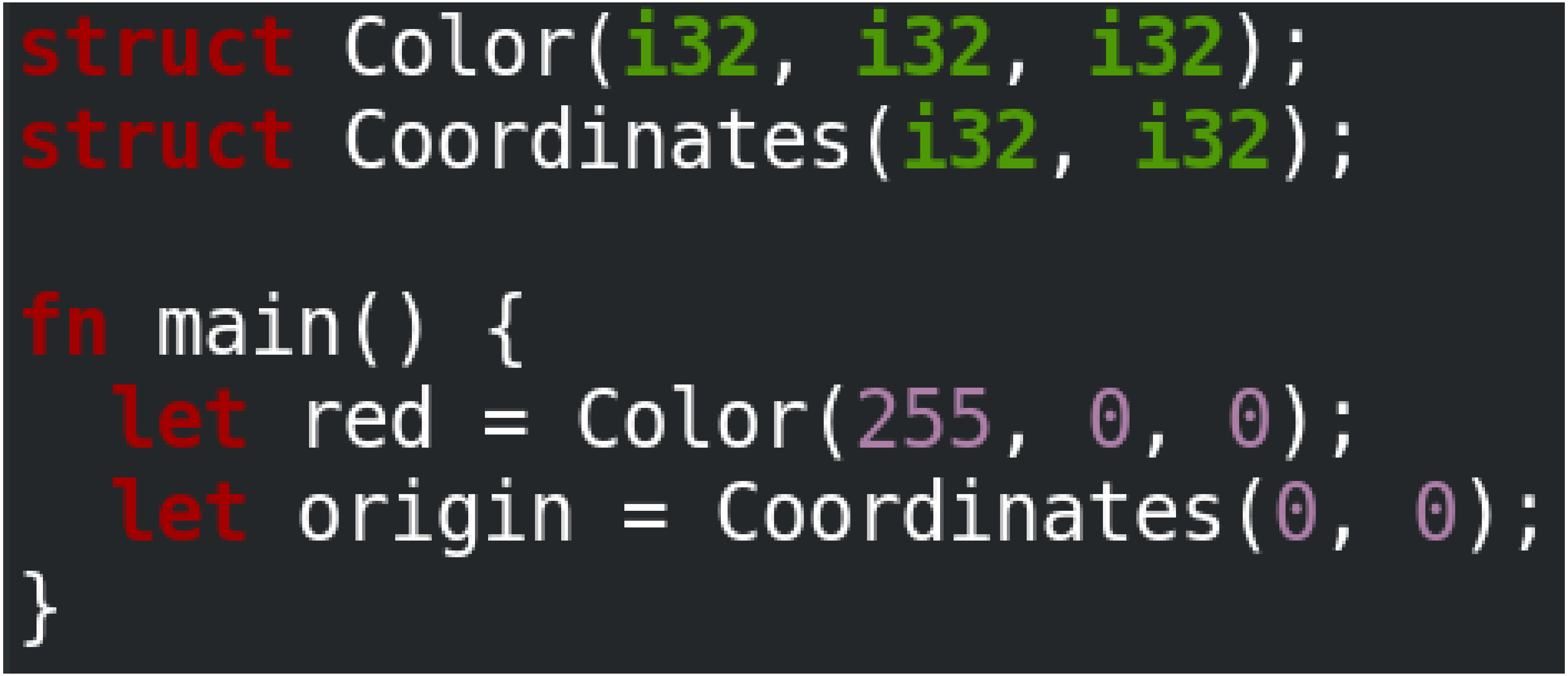
```rust
fn create_person() -> Person {
  Person {
    age: 25,
    firstname: String::from("Max"),
    lastname: String::from("Mustermann"),
  }
}
```

Instantiating Structs (3)



```rust
fn create_person(age: i32) -> Person {
  Person {
    age,
    firstname: String::from("Max"),
    lastname: String::from("Mustermann"),
  }
}
```

Tuple Structs

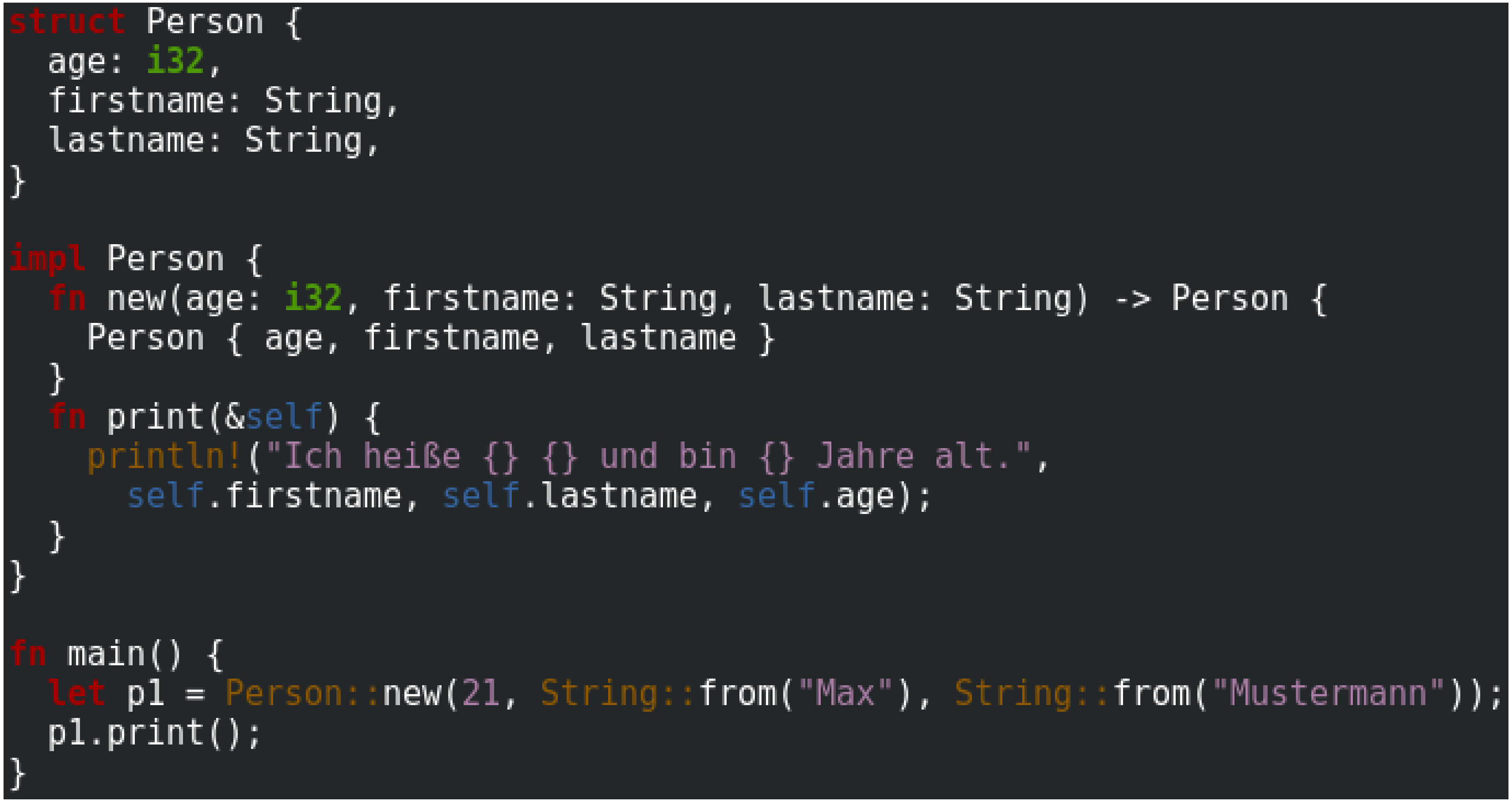


```rust
struct Color(i32, i32, i32);
struct Coordinates(i32, i32);

fn main() {
  let red = Color(255, 0, 0);
  let origin = Coordinates(0, 0);
}
```

The impl block

- Used to define methods on structs.
- Example: `String::len()` is method `len()` on struct `String`
- Chap. 10: used to implement `traits`.

The impl block (2)



```rust
struct Person {
  age: i32,
  firstname: String,
  lastname: String,
}

impl Person {
  fn new(age: i32, firstname: String, lastname: String) -> Person {
    Person { age, firstname, lastname }
  }
  fn print(&self) {
    println!("Ich heiße {} {} und bin {} Jahre alt.",
      self.firstname, self.lastname, self.age);
  }
}

fn main() {
  let p1 = Person::new(21, String::from("Max"), String::from("Mustermann"));
  p1.print();
}
```

Thank you

- Information: The Rust Programming Language, Chapter 5
- Code examples: Rust Book, slightly adjusted by myself
- Images: Screenshots taken by myself