**Технологии конструирования программного обеспечения**

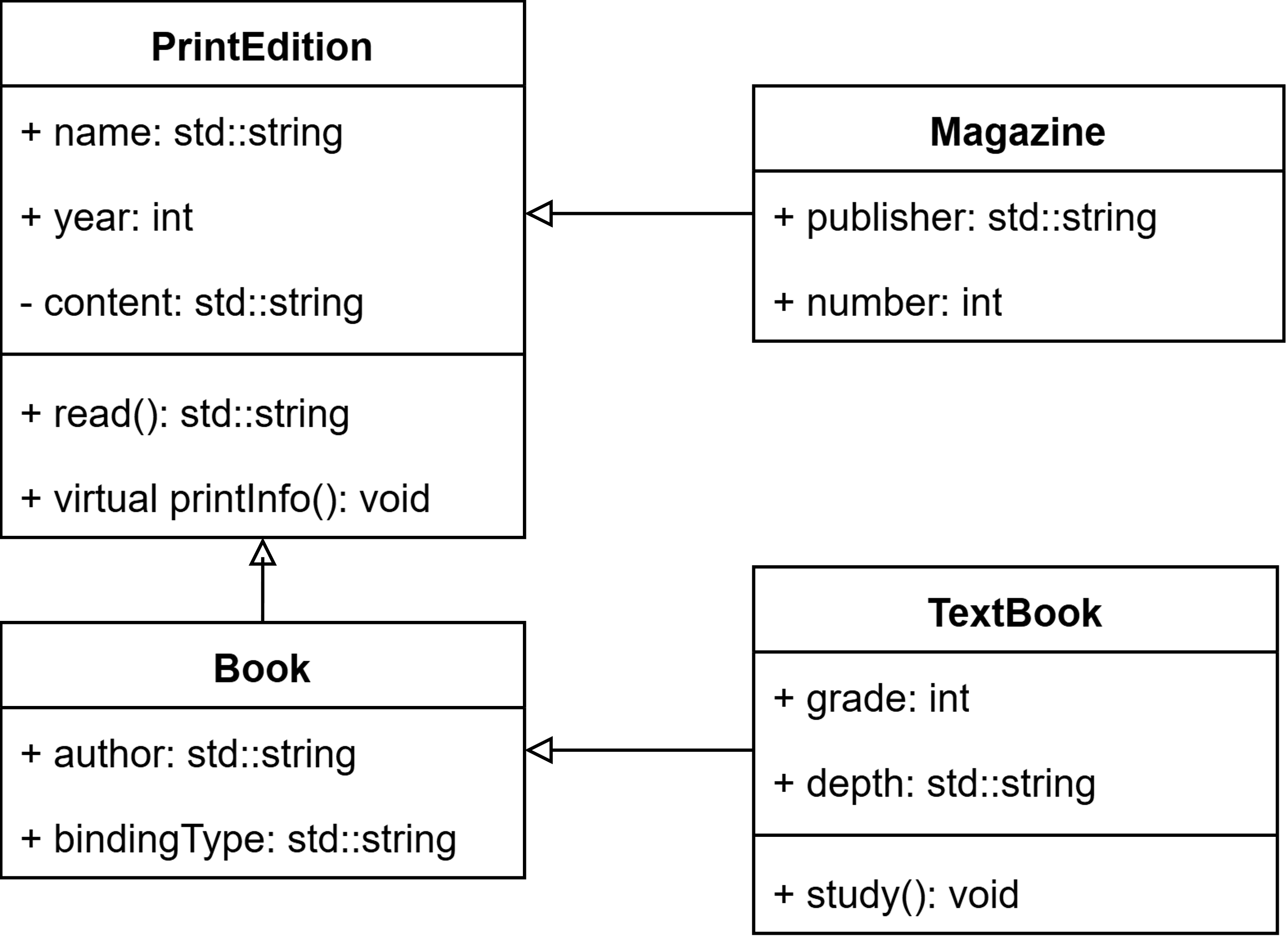
**Отчет по лабораторной работе № 2**

**Группа:** 221-3210 **Студент:** Обухов Алексей Сергеевич

**Задание на лабораторную работу**

3. Журнал, книга, печатное издание, учебник.

**Диаграмма классов**

****

**Диаграмма последовательности**

**Исходный код программы**

Файл «PrintEditions.h»

#include <string>

#pragma once

// Базовый класс для печатных изданий

class PrintEdition {

protected:

// Содержимое издания

std::string content;

public:

std::string name; // Название

int year; // Год выпуска

PrintEdition() {

// Значения используются как заглушки

this->content = "NoContent";

this->name = "NoName";

this->year = 0;

};

PrintEdition(std::string name, int year, std::string content) {

this->name = name;

this->year = year;

this->content = content;

}

// Функция для получения содержимого печатного издания

std::string read() {

return this->content;

}

// Функция для вывода информации о печатном издании

// Для классов-потомков переопределяется для

// соответствия их атрибутам

virtual void printInfo() {

std::cout << "Тип: " << typeid(\*this).name() << "\n";

std::cout << "Название: " << this->name << "\n";

std::cout << "Год: " << this->year << "\n";

}

};

class Magazine : public PrintEdition {

public:

// Издатель журнала

std::string publisher = "NoPublisher";

// Номер журнала

int number = 0;

Magazine(std::string name, int year, std::string content, std::string publisher, int number) {

this->name = name;

this->year = year;

this->content = content;

this->publisher = publisher;

this->number = number;

}

virtual void printInfo() {

std::cout << "Тип: " << typeid(\*this).name() << "\n";

std::cout << "Название: " << this->name << "\n";

std::cout << "Год: " << this->year << "\n";

std::cout << "Издатель: " << this->publisher << "\n";

std::cout << "Номер: " << this->number << "\n";

}

};

class Book : public PrintEdition {

public:

std::string author = "NoAuthor"; // Автор книги

std::string bindingType = "NoBindingType"; // Тип переплета (например: жесткий, мягкий)

Book() {};

Book(std::string name, int year, std::string content, std::string author, std::string bindingType) {

this->name = name;

this->year = year;

this->content = content;

this->author = author;

this->bindingType = bindingType;

}

virtual void printInfo() {

std::cout << "Тип: " << typeid(\*this).name() << "\n";

std::cout << "Название: " << this->name << "\n";

std::cout << "Год: " << this->year << "\n";

std::cout << "Автор: " << this->author << "\n";

std::cout << "Переплет: " << this->bindingType << "\n";

}

};

class TextBook : public Book {

public:

int grade // Класс, на который рассчитан учебник

std::string depth; // Уровень учебника (например: базовый, углубленный)

TextBook(std::string name, int year, std::string content, std::string author, std::string bindingType, int grade, std::string depth) {

this->name = name;

this->year = year;

this->content = content;

this->author = author;

this->bindingType = bindingType;

this->grade = grade;

this->depth = depth;

}

virtual void printInfo() {

std::cout << "Тип: " << typeid(\*this).name() << "\n";

std::cout << "Название: " << this->name << "\n";

std::cout << "Год: " << this->year << "\n";

std::cout << "Автор: " << this->author << "\n";

std::cout << "Переплет: " << this->bindingType << "\n";

std::cout << "Класс: " << this->grade << "\n";

std::cout << "Уровень: " << this->depth << "\n";

}

};

Файл «main.cpp»

#include <iostream>

#include <string>

#include "PrintEditions.h"

// Фукнция для чтения изданий а также вывода информации о них

void read(PrintEdition\* printEdition) {

printEdition->printInfo();

std::cout << "\nСодержимое:\n";

std::cout << printEdition->read() << "\n\n";

}

int main() {

setlocale(LC\_ALL, "Russian");

PrintEdition printEdition(std::string("Печатное издание"), 2015, std::string("Содержимое печатного издания"));

Magazine magazine(std::string("Журнал"), 2016, std::string("Содержимое журнала"), std::string("Издатель журнала"), 123);

Book book(std::string("Книга"), 2017, std::string("Содержимое книги"), std::string("Автор книги"), std::string("Жесткий переплет"));

TextBook textbook(std::string("Учебник"), 2018, std::string("Содержимое учебника"), std::string("Автор учебника"),

std::string("Жесткий переплет"), 10, std::string("Базовый уровень"));

read(&printEdition);

read(&magazine);

read(&book);

read(&textbook);

return 0;

}