Вступление

C#

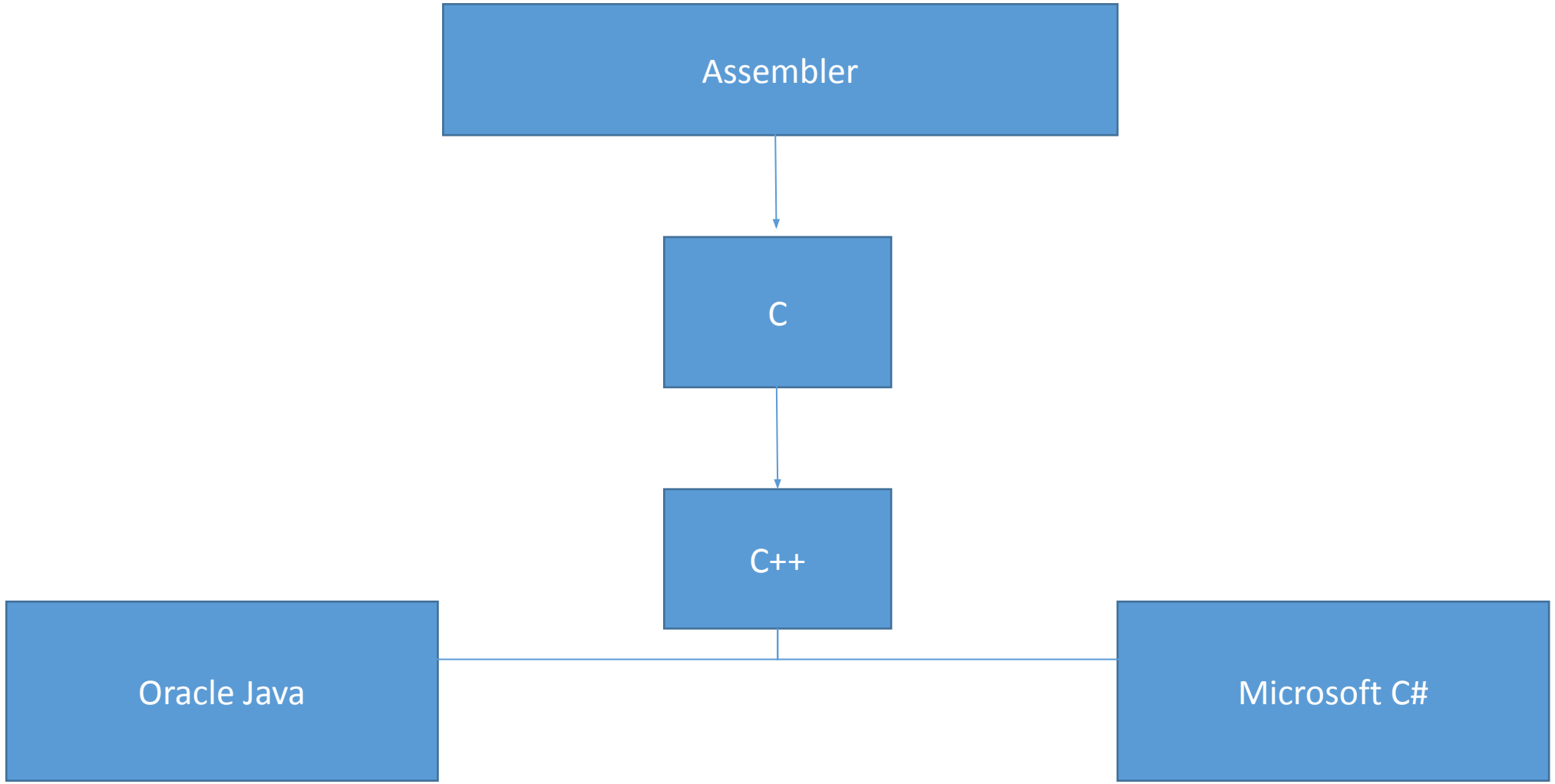
Области применения C#

- Десктопные приложения Win
- Банковские приложения
- Мобильные приложения (Xamarin.Forms, UWP)
- Игры (Unity)
- Кроссплатформенные приложения
- Работа с БД
- WEB
- ПО для серверов на базе Microsoft Windows Platform
- Работа с облаком (Azure)

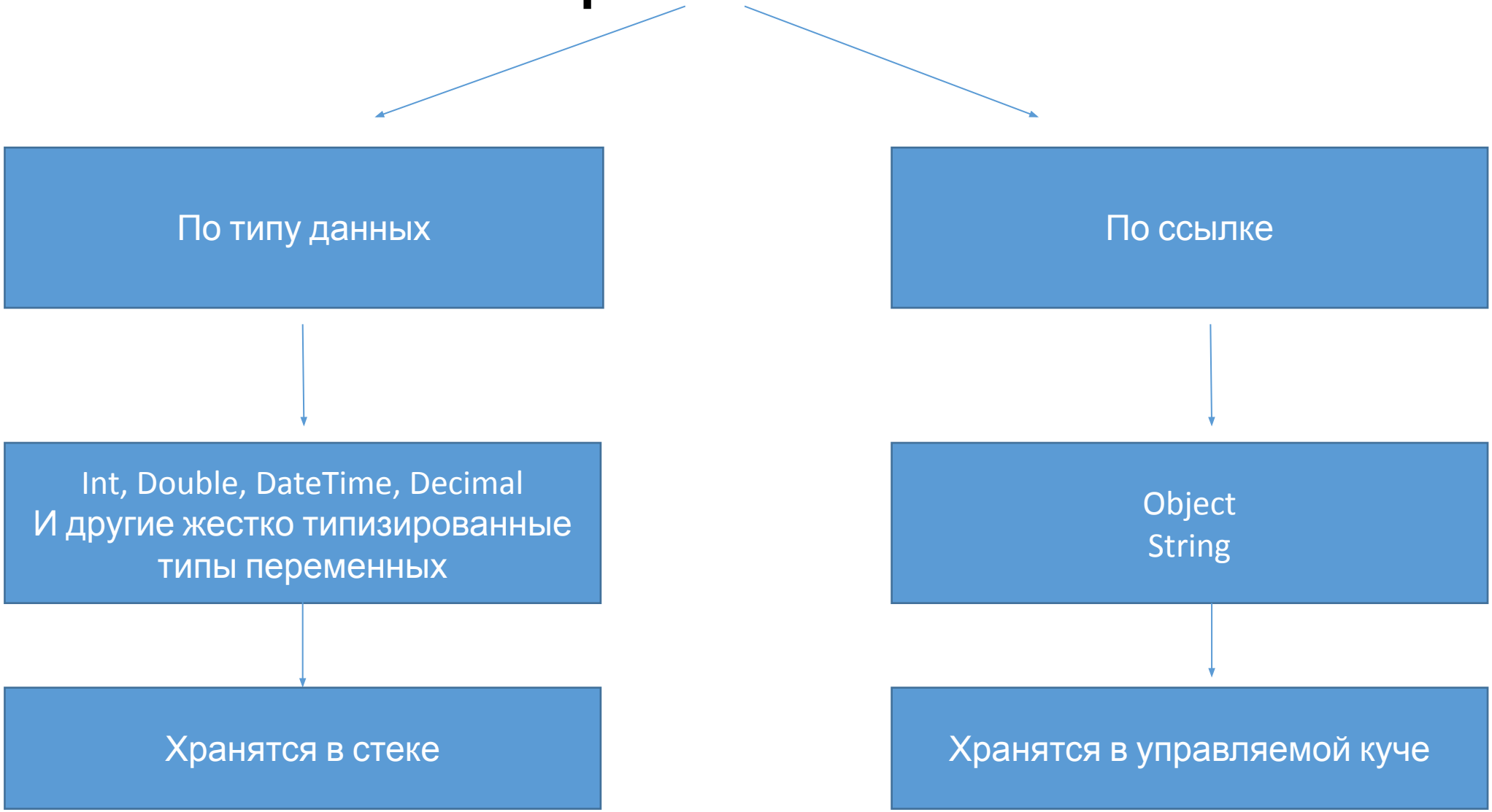
Почему стоит начать изучать C#

- Простота освоения
- Понятная и подробная документация (MSDN)
- Интуитивно понятный синтаксис
- Огромное количество синтаксического сахара
- C# - мощный инструмент для решения огромного ряда задач
- Шикарная перспектива монетизации знаний
- Поддержка большого ряда устройств (Xamarin.Forms, UWP)
- Большое сообщество разработчиков и огромное количество ресурсов для самостоятельного освоения языка

История создания (Архитектура)



Переменные


По типу данных
По ссылке
Int, Double, DateTime, Decimal
И другие жестко типизированные
типы переменных
Object
String
Хранятся в стеке
Хранятся в управляемой куче

Определение своего типа данных

```
struct MyTypeOfVariable
{
    public int Num;
    public string Id;
}
```

Методы

Метод ~ Функция/Процедура

- Методу необходим контейнер, к которому он будет привязан (например, Класс)
- Метод может возвращать значение
- Метод может не возвращать значение (возвращать void)

Определение метода

```
public void MyNewMethod()
{

}
```

Модификатор_доступа Тип_возвращаемого_значения Название (входные переменные)
{
return переменная_возвращаемого_типа;
}

Модификаторы доступа

- public: публичный, общедоступный класс или член класса. Такой член класса доступен из любого места в коде, а также из других программ и сборок.
- private: закрытый класс или член класса. Представляет полную противоположность модификатору public. Такой закрытый класс или член класса доступен только из кода в том же классе или контексте.
- protected: такой член класса доступен из любого места в текущем классе или в производных классах.
- internal: класс и члены класса с подобным модификатором доступны из любого места кода в той же сборке, однако он недоступен для других программ и сборок (как в случае с модификатором public).
- protected internal: совмещает функционал двух модификаторов. Классы и члены класса с таким модификатором доступны из текущей сборки и из производных классов.

Классы

Класс — это логическая структура(контейнер), позволяющая создавать свои собственные пользовательские типы путем группирования переменных других типов, методов и событий.

```
public class Student
{
    text
}
```

МД class Название { #переменные,события,методы# }

Конструктор класса

Конструкторы экземпляров используются для создания переменных члена экземпляра, если создание объекта class осуществляется с помощью выражения new

```
public class NewClassForExample
{
    ссылок: 0
    public NewClassForExample()
    {

    }
}
```

Конструктор

Наследование классов

```
public class Nerd : Student
{
    ссылок: 0
    public Nerd(string name, int age, string faqName = "", int course = 2)...

    ссылок: 0
    public void ChangeFaq(string nextFaq)...
}
```

Литература и полезные ссылки

- Герберт Шилдт - C# 4.0. Полное руководство
- http://losev-al.blogspot.ru/
- https://www.youtube.com/user/Defazze
- Курс увлекательного программирования на C# от Сошникова
- Google.com (хороший прогер сначала гуглит)
- Джеффри Рихтер – ”CLR via C#, Framework 4.5”