

3. Where 操作符

Where 操作符，在第一章第二章都有进行接触。这节课主要是实践一下上两堂课所说的格式规范。
也就是介绍一个操作符的时候，先介绍 LINQ 这种方式。

LINQ Where 简介:

LINQ 中的 Where 操作符与 SQL 命令中的 Where 作用相似，都是起到范围限定也就是过滤作用的，而条件判断条件就是它后面所接的子句。

LINQ Where 示例代码:

查询大于 45 岁的学生:

```
/******  
 * http://sikiedu.com liangxie  
*****/  
  
using System.Collections.Generic;  
using System.Linq;  
using UnityEngine;  
  
namespace UniRxLesson  
{  
    public class LINQWhereExample : MonoBehaviour  
    {  
        class Student  
        {  
            public string Name;  

```

```
        public int Age;
    }

    private void Start()
    {
        var students = new List<Student>()
        {
            new Student() {Name = "张三", Age = 50},
            new Student() {Name = "张三", Age = 50},
            new Student() {Name = "李四", Age = 40}
        };

        var oldStudents = students.Where(student => student.Age > 45);

        foreach (var student in oldStudents)
        {
            Debug.Log(student.Name);
        }
    }
}
```

```
}  
}  
}
```

输出结果为:

```
张三  
张三
```

在这里呢，要介绍一个概念，叫做查询语句/句法。

LINQ Where 查询式

以上的代码，改成查询句式后，代码如下:

```
/*  
* http://sikiedu.com liangxie  
*/
```

```
using System.Collections.Generic;
using System.Linq;
using UnityEngine;

namespace UniRxLesson
{

    public class LINQWhereQueryExample : MonoBehaviour
    {
        class Student
        {
            public string Name;

            public int Age;
        }

        // Use this for initialization
```

```
void Start()
{
    var students = new List<Student>()
    {
        new Student() {Name = "张三", Age = 50},
        new Student() {Name = "张三", Age = 50},
        new Student() {Name = "李四", Age = 40}
    };

    var oldStudents = from oldStudent in students
                      where oldStudent.Age > 45
                      select oldStudent;

    foreach (var student in oldStudents)
    {
        Debug.Log(student.Name);
    }
}
```

```
}
```

代码非常容易理解。

输出的结果，与前边一样：

```
张三
```

```
张三
```

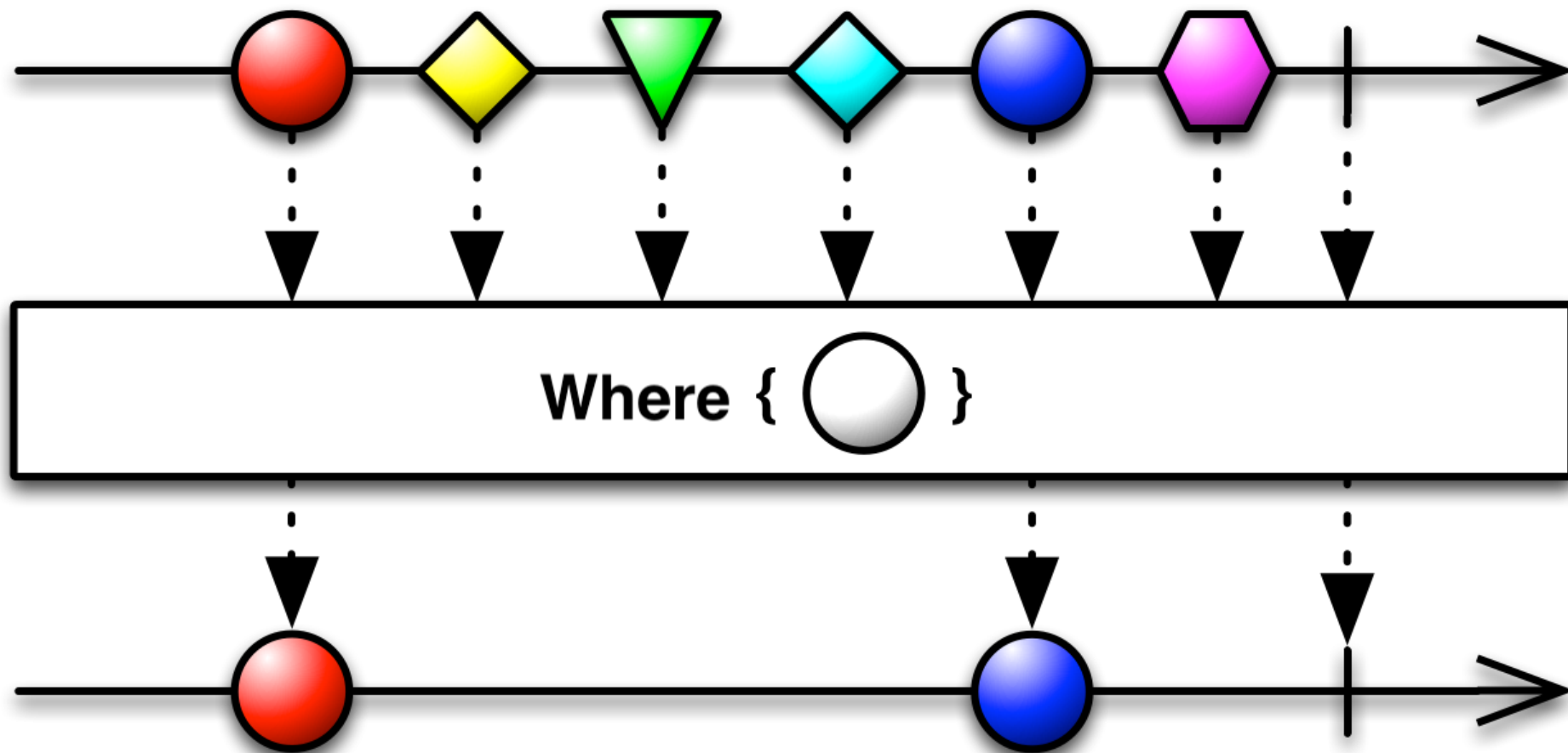
UniRx Where 示意图



Where ($x \Rightarrow x > 10$)

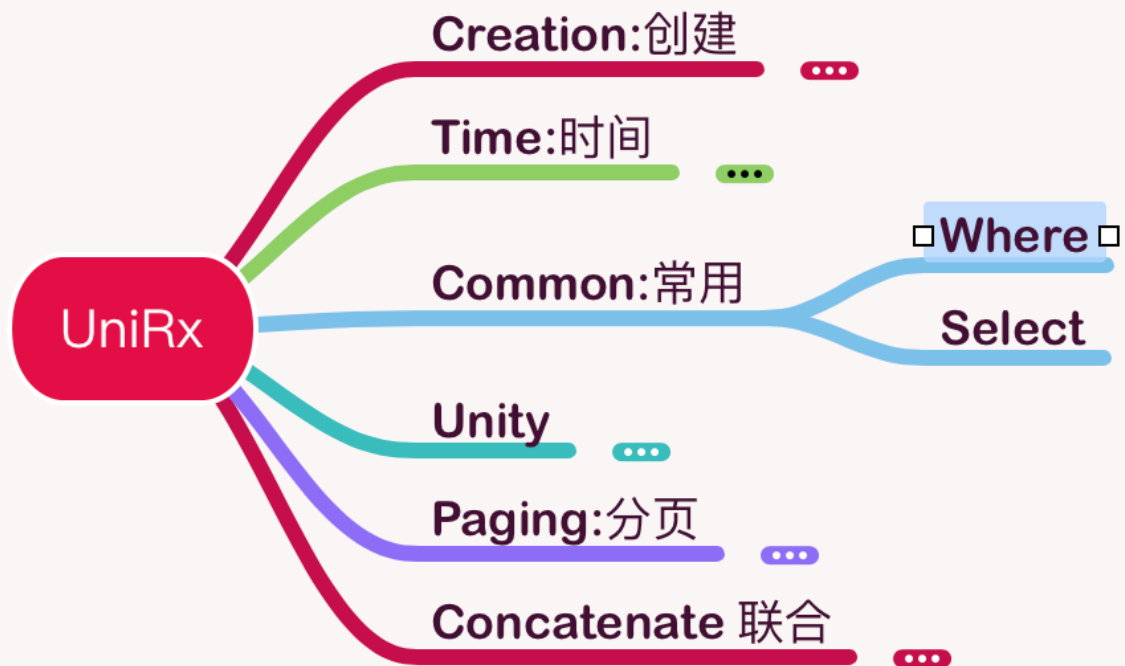


只有满足条件的通过。



只有圆形的通过。

UniRx Where 在知识地图中的位置



它是非常常用的操作符。

UniRx Where 示例代码

鼠标点击事件监听.

```
/*  
 * http://sikiedu.com liangxie  
 */  
  
using UniRx;  
using UnityEngine;  
  
namespace UniRxLesson  
{  
    public class UniRxWhereExamlle : MonoBehaviour  
    {  
        // Use this for initialization
```

```
private void Start()
{
    Observable.EveryUpdate()
        .Where(_ => Input.GetMouseButtonDown(0))
        .Subscribe(_ => { Debug.Log("mouse down"); })
        .AddTo(this);
}
}
```

当每次点击鼠标左键的时候，就会输出 mouse down。

当然，UniRx 也是支持查询式语法的。

UniRx Where 查询式:

改成查询式如下

```
/******
```

```
* http://sikiedu.com liangxie
```

```
*****/
```

```
using UniRx;
```

```
using UnityEngine;
```

```
namespace UniRxLesson
```

```
{
```

```
    public class UnIRxWhereQueryExample : MonoBehaviour
```

```
    {
```

```
        private void Start()
```

```
        {
```

```
            var mouseClickEventStreams = from updateEvent in Observable.EveryUpdate()
```

```
                where Input.GetMouseButtonDown(0)
```

```
                select updateEvent;
```

```
            mouseClickEventStreams
```

```
                .Subscribe(_ => { Debug.Log("mouse clicked"); })
```

```
                .AddTo(this);
```



```
}  
}  
}
```

结果和上述代码一样。

小结:

今天就按照在前几堂课制定的规范，讲解了一下 Where 操作符，希望同学们习惯这种方式，在之后，只要有新的操作符，就通过这种方式进行学习。

OK，今天的内容就这些。