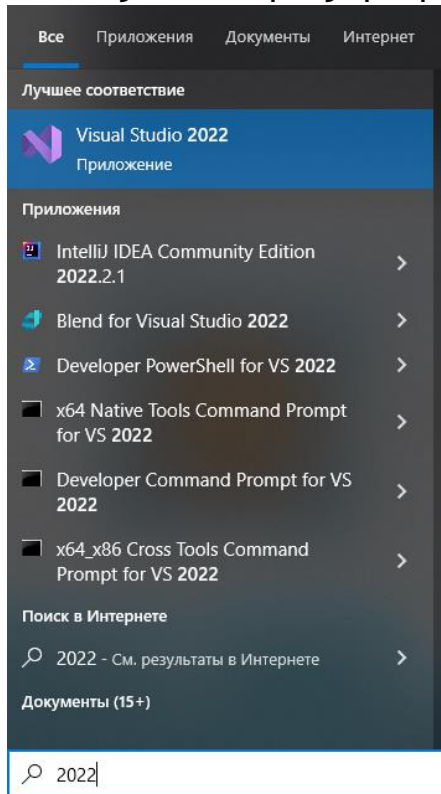


## 1. Создание приложения

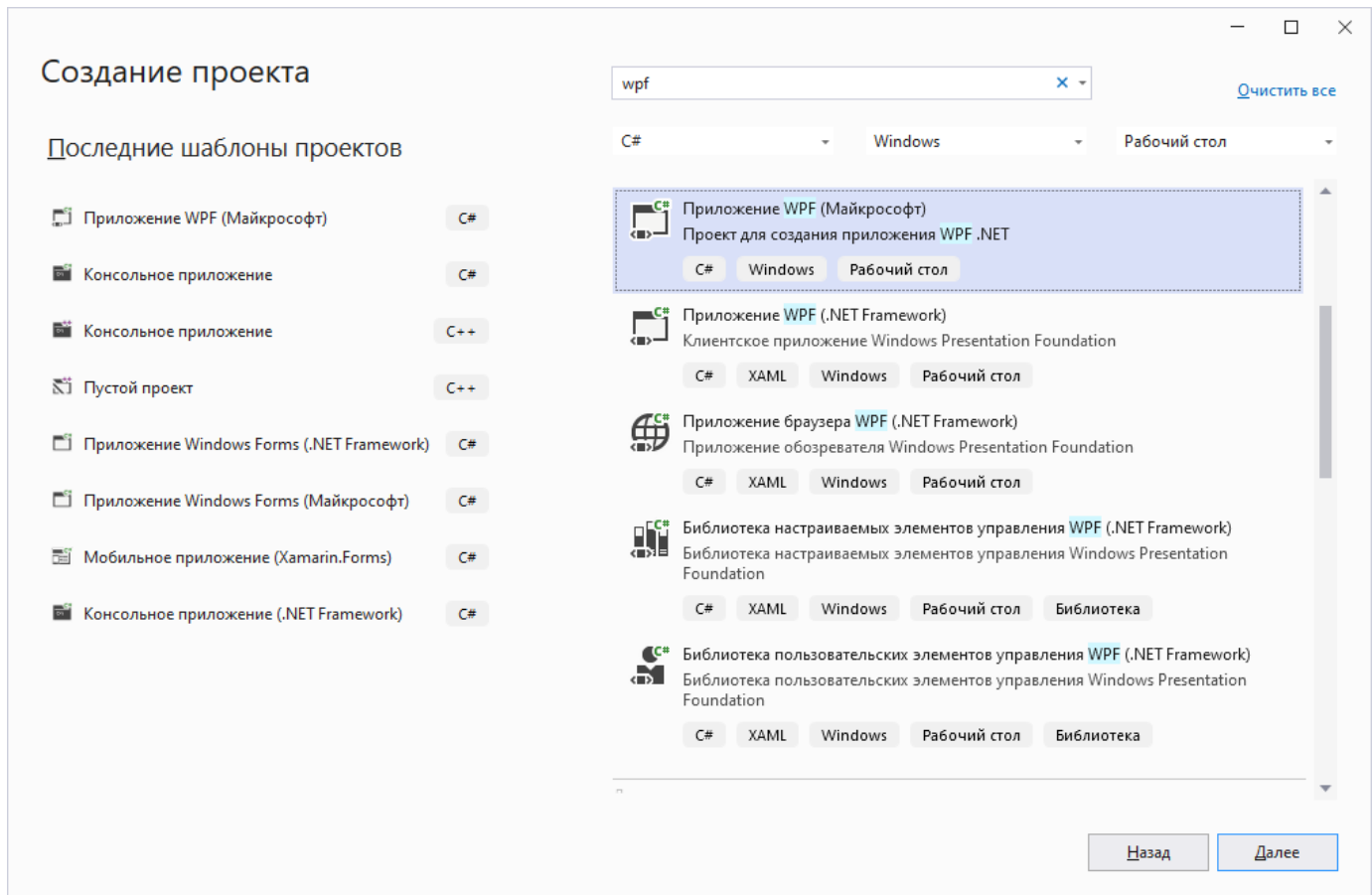
### 1) Запустить среду разработки Visual Studio 2022



### 2) Нажать «Создание проекта»

### 3) Выбрать программирования C#, в строке поиска написать wpf

### 4) Из списка проектов выбрать «Приложение WPF (Майкрософт)»



### 5) В открывшемся окне можно оставить настройки по умолчанию и нажать кнопку «Создать».

## 2. Настройка интерфейса

1) В открывшемся окне вместо

`<Grid>`

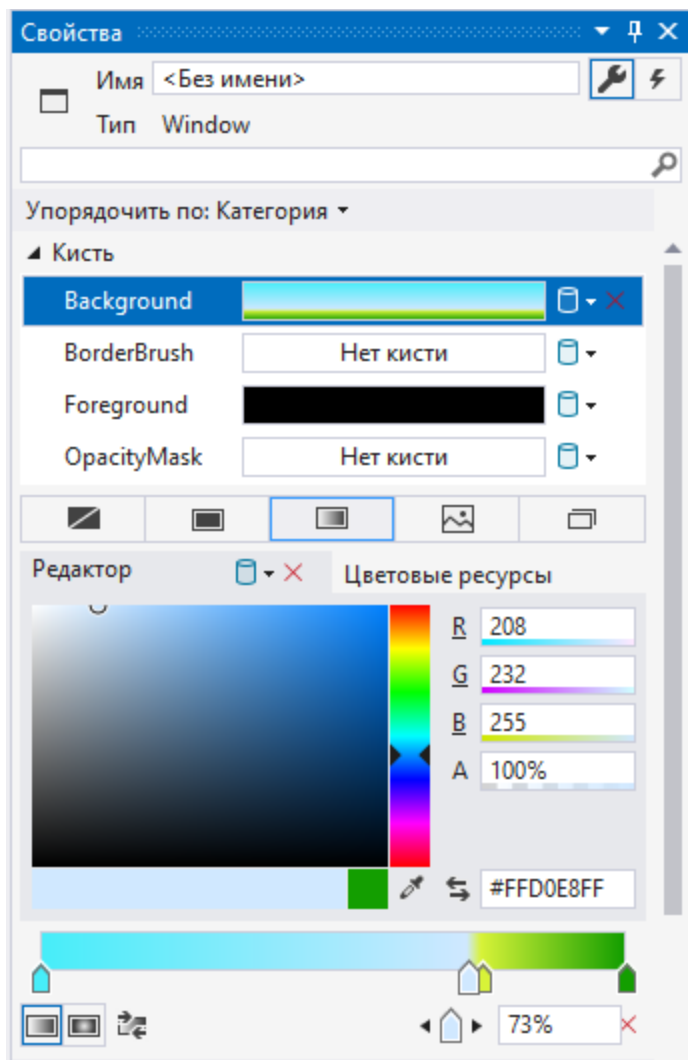
`</Grid>`

написать:

`<Canvas x:Name="gameCanvas" />`

2) Во вкладке «Свойства» настроить фон игрового поля:

- нажать на надпись «Кисть»
- выбрать Background
- выбрать градиентную кисть
- выбрать любые цвета для градиентной заливки (можно выбрать 2 и более)



### 3. Генерация шариков

1) Нажать F7, чтобы открылось окно с кодом:

```
public partial class MainWindow : Window
{
    Ссылка 0
    public MainWindow()
    {
        InitializeComponent();
    }
}
```

2) Создать таймер, добавив подчеркнутые строки:

DispatcherTimer timer = new DispatcherTimer(DispatcherPriority.Render);

```
public MainWindow()
{
    InitializeComponent();

    timer.Interval = TimeSpan.FromMilliseconds(500);
    timer.Start();
    timer.Tick += //после += нажать Tab для появления Timer_Tick
}
```

3) В появившемся Timer\_Tick надо изменить код на подчеркнутые строки:

```
private void Timer_Tick(object? sender, EventArgs e)
{
    // программно добавить врага
    Ellipse enemy = new Ellipse
    {
        Width = 30,
        Height = 30,
        Fill = new SolidColorBrush(Colors.Purple),
    };
    gameCanvas.Children.Add(enemy);

    // разместить врага в случайной точке
    double x = new Random().NextDouble() * Width;
    Canvas.SetLeft(enemy, x);
    Canvas.SetTop(enemy, -enemy.Height/2);

    // добавить обработчик нажатия на врага
    enemy.MouseDown += //после += нажать Tab для появления Enemy_MouseDown
}
```

4) В появившемся Enemy\_MouseDown надо изменить код на подчеркнутые строки:

```
private void Enemy_MouseDown(object sender, MouseButtonEventArgs e)
{
    // удаление врага при попадании
    gameCanvas.Children.Remove(sender as UIElement);
}
```

#### 4. Добавление движения

1) Создать второй таймер timerMove, добавив в код подчеркнутые строки:

```
DispatcherTimer timer = new DispatcherTimer(DispatcherPriority.Render);  
DispatcherTimer timerMove = new DispatcherTimer(DispatcherPriority.Render);
```

```
public MainWindow()  
{  
    InitializeComponent();  
  
    timer.Interval = TimeSpan.FromMilliseconds(500);  
    timer.Start();  
    timer.Tick += Timer_Tick;  
  
timerMove.Interval = TimeSpan.FromMilliseconds(50);  
timerMove.Start();  
timerMove.Tick += //после += нажать Tab для появления TimerMove_Tick  
}
```

2) В появившемся TimerMove\_Tick надо изменить код на подчеркнутые строки:

```
private void Timer_Tick(object? sender, EventArgs e)  
{  
    foreach (Shape item in gameCanvas.Children.OfType<Shape>())  
    {  
        double y = (double)item.GetValue(Canvas.TopProperty);  
        Canvas.SetTop(item, y + 3);  
  
        if (y >= 400)  
        {  
            timerMove.Stop();  
            MessageBox.Show("End");  
            break;  
        }  
    }  
}
```

## 5. Отображение счёта

1) Создать переменную `score` и метку `scoreLabel` для отображения счёта, добавив в код подчеркнутые строки:

```
DispatcherTimer timer = new DispatcherTimer(DispatcherPriority.Render);
DispatcherTimer timerMove = new DispatcherTimer(DispatcherPriority.Render);
```

```
int score = 0;
Label scoreLabel = new Label();
```

```
public MainWindow()
{
    InitializeComponent();

    timer.Interval = TimeSpan.FromMilliseconds(500);
    timer.Start();
    timer.Tick += Timer_Tick;

    timerMove.Interval = TimeSpan.FromMilliseconds(50);
    timerMove.Start();
    timerMove.Tick += TimerMove_Tick;

    gameCanvas.Children.Add(scoreLabel);
    scoreLabel.Foreground = new SolidColorBrush(Colors.White);
    scoreLabel.Content = 0;
}
```

2) В `Enemy_MouseDown` добавить в код подчеркнутые строки:

```
private void Enemy_MouseDown(object sender, MouseButtonEventArgs e)
{
    // удаление врага при попадании
    gameCanvas.Children.Remove(sender as UIElement);

    // вывод результата на экран
    score = score + 1;
    scoreLabel.Content = score;
}
```