

Справочник

Справочник

10.07.2024

Яндекс

Справочник. Версия 1.0

Дата подготовки документа: 10.07.2024

Этот документ является составной частью технической документации Яндекса.

© 2008—2024 ООО «ЯНДЕКС». Все права защищены.

Предупреждение об исключительных правах и конфиденциальной информации

Исключительные права на все результаты интеллектуальной деятельности и приравненные к ним средства индивидуализации юридических лиц, товаров, работ, услуг и предприятий, которым предоставляется правовая охрана (интеллектуальную собственность), используемые при разработке, поддержке и эксплуатации службы, включая, но не ограничиваясь, программы для ЭВМ, базы данных, изображения, тексты, другие произведения, а также изобретения, полезные модели, товарные знаки, знаки обслуживания, коммерческие обозначения и фирменные наименования, принадлежат ООО «ЯНДЕКС» либо его лицензиарам.

Использование результатов интеллектуальной деятельности и приравненных к ним средств индивидуализации в целях, не связанных с разработкой, поддержкой и эксплуатацией службы, не допускается без получения предварительного согласия правообладателя. Настоящий документ содержит конфиденциальную информацию ООО «ЯНДЕКС». Использование конфиденциальной информации в целях, не связанных с разработкой, поддержкой и эксплуатацией службы, а равно как и разглашение таковой, не допускается. При этом под разглашением понимается любое действие или бездействие, в результате которых конфиденциальная информация в любой возможной форме (устной, письменной, иной форме, в том числе с использованием технических средств) становится известной третьим лицам без согласия обладателя такой информации либо вопреки трудовому или гражданско-правовому договору.

Отношения ООО «ЯНДЕКС» с лицами, привлекаемыми для разработки, поддержки и эксплуатации службы, регулируются законодательством Российской Федерации и заключаемыми в соответствии с ним трудовыми и/или гражданско-правовыми договорами (соглашениями). Нарушение требований об охране результатов интеллектуальной деятельности и приравненных к ним средств индивидуализации, а равно как и конфиденциальной информации, влечет за собой дисциплинарную, гражданско-правовую, административную или уголовную ответственность в соответствии с законодательством Российской Федерации.

Контактная информация

ООО «ЯНДЕКС»

<https://www.yandex.ru>

Тел.: +7 495 739 7000

Email: pr@yandex-team.ru

Главный офис: 119021, Россия, г. Москва, ул. Льва Толстого, д. 16

Содержание

AdRequest.Builder	5
Описание методов.....	5
AdRequest	7
Описание свойств.....	7
AdSize	8
Описание методов.....	9
Описание свойств.....	10
Описание полей.....	10
Banner	11
Описание методов.....	11
Описание событий.....	12
Gender	13
Описание констант.....	13
Interstitial	13
Описание методов.....	14
Описание событий.....	15
Location.Builder	16
Описание методов.....	16
Location	17
Описание свойств.....	18
MobileAds	18
Описание методов.....	19
RewardedAd	19
Описание методов.....	20
Описание событий.....	21
Reward	22
Описание методов.....	22
Описание полей.....	23
ScreenUtils	23
Описание методов.....	23

AdPosition.....	23
Описание перечислений.....	24
AdSizeType.....	24
Описание перечислений.....	25

Класс AdRequest.Builder

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс для построения таргетированного запроса рекламы.

Методы

Builder WithAge(string age)	Задает возраст пользователя в приложении.
Builder WithContextQuery(string contextQuery)	Задает поисковый запрос, который был введен пользователем внутри приложения.
Builder WithContextTags(List<string> contextTags)	Задает список тегов, которые описывают текущий контекст пользователя. Список зависит от того, на каком экране находится пользователь.
Builder WithGender(string gender)	Задает пол пользователя в приложении.
Builder WithLocation(Location location)	Задает пользовательскую локацию.
Builder WithParameters(Dictionary<string, string> parameters)	Задает список дополнительных параметров.
Builder WithAdRequest(AdRequest adRequest)	Задает AdRequest.
AdRequest Build()	Создает объект класса AdRequest с данными для таргетированного запроса рекламы.

Описание методов

Builder WithAge(string age)

```
public Builder WithAge(string age)
```

Задает возраст пользователя в приложении.

Параметры:

age	Возраст пользователя.
-----	-----------------------

Возвращает:

Объект класса AdRequest.Builder с указанным возрастом пользователя.

Builder WithContextQuery(string contextQuery)

```
public Builder WithContextQuery(string contextQuery)
```

Задает поисковый запрос, который был введен пользователем внутри приложения.

Параметры:

contextQuery	Поисковый запрос, который был введен пользователем внутри приложения.
--------------	---

Возвращает:

`adRequest`Объект класса `AdRequest`.**Возвращает:**Объект класса `AdRequest.Builder`, созданный на основе переданного объекта класса `AdRequest`.**AdRequest Build()**

```
public AdRequest Build()
```

Создает объект класса `AdRequest` с данными для таргетированного запроса рекламы.**Параметры:**`adRequest`Объект класса `AdRequest`.**Возвращает:**Объект класса `AdRequest` с данными для таргетированного запроса рекламы.

Класс AdRequest

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс содержит настройки для получения таргетированного рекламного объявления. Для построения запроса используйте [Класс AdRequest.Builder](#).

Свойства`string Age { get }`

Возраст пользователя, который был введен в приложении.

.

`string ContextQuery { get }`

Поисковый запрос, который был введен пользователем в приложении.

.

`List<string> ContextTags { get }`

Список тегов, которые выставляет приложение.

.

`string Gender { get }`

Пол пользователя, который был введен в приложении.

.

`Location Location { get }`

Пользовательская локация.

.

`Dictionary<string, string> Parameters { get }`

Список дополнительных параметров.

.

Описание свойств

`string Age { get }`

```
public string Age { get }
```

Возраст пользователя, который был введен в приложении.

string ContextQuery { get }

```
public string ContextQuery { get }
```

Поисковый запрос, который был введен пользователем в приложении.

List<string> ContextTags { get }

```
public List<string> ContextTags { get }
```

Список тегов, которые выставляет приложение.

string Gender { get }

```
public string Gender { get }
```

Пол пользователя, который был введен в приложении.

Location Location { get }

```
public Location Location { get }
```

Пользовательская локация.

Dictionary<string, string> Parameters { get }

```
public Dictionary<string, string> Parameters { get }
```

Список дополнительных параметров.

Класс AdSize

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс отвечает за размеры баннера.

Методы

`static AdSize StickySize (int width)`

Создает объект класса AdSize с заданной шириной «sticky» баннера. Высота баннера зависит от того, какие типы блоков выбраны в Партнерском интерфейсе.

`static AdSize FlexibleSize (int width, int height)`

Создает объект класса AdSize с заданной максимальной высотой и шириной баннера. Высота баннера зависит от того, какие типы блоков выбраны в Партнерском интерфейсе.

Свойства

`int Width { get }`

Ширина баннера.

`int Height { get }`

Высота баннера.

`AdSizeType AdSizeType { get }`

Размеры баннера.

Поля

<code>public static readonly AdSize BANNER_240x400 (deprecated)</code>	Баннер с размерами 240x400 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_300x250 (deprecated)</code>	Баннер с размерами 300x250 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_300x300 (deprecated)</code>	Баннер с размерами 300x300 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_320x50 (deprecated)</code>	Баннер с размерами 320x50 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_320x100 (deprecated)</code>	Баннер с размерами 320x100 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_400x240 (deprecated)</code>	Баннер с размерами 400x240 в dp (density-independent pixels).
<code>public static readonly AdSize BANNER_728x90 (deprecated)</code>	Баннер с размерами 728x90 в dp (density-independent pixels).

Описание методов**StickySize(int width)**

```
static AdSize YandexMobileAds.Base.AdSize.StickySize (int width)
```

Создает объект класса AdSize с заданной шириной «sticky» баннера. Высота баннера зависит от того, какие типы блоков выбраны в Партнерском интерфейсе.

Параметры:

width	Ширина «sticky» баннера.
-------	--------------------------

Возвращает:

Объект класса AdSize с заданной шириной «sticky» баннера.

FlexibleSize (int width, int height)

```
static AdSize YandexMobileAds.Base.AdSize.FlexibleSize (int width, int height)
```

Создает объект класса AdSize с заданной максимальной высотой и шириной баннера. Высота баннера зависит от того, какие типы блоков выбраны в Партнерском интерфейсе.

Параметры:

width	Максимальная ширина баннера.
height	Максимальная высота баннера.

Возвращает:

Объект класса AdSize с заданной максимальной шириной и высотой баннера

Описание свойств

int Width { get }

```
public int Width { get }
```

Ширина баннера.

int Height { get }

```
public int Height { get }
```

Высота баннера.

int Height { get }

```
public AdSizeType AdSizeType { get }
```

Размеры баннера.

Описание полей

public static readonly AdSize BANNER_240x400 (deprecated)

```
public static readonly AdSize BANNER_240x400 = new AdSize {Width = 240, Height = 400, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 240x400 в dp (density-independent pixels).

public static readonly AdSize BANNER_300x250 (deprecated)

```
public static readonly AdSize BANNER_300x250 = new AdSize {Width = 300, Height = 250, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 300x250 в dp (density-independent pixels).

public static readonly AdSize BANNER_300x300 (deprecated)

```
public static readonly AdSize BANNER_300x250 = new AdSize {Width = 300, Height = 300, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 300x300 в dp (density-independent pixels).

public static readonly AdSize BANNER_320x50 (deprecated)

```
public static readonly AdSize BANNER_320x50 = new AdSize {Width = 320, Height = 50, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 320x50 в dp (density-independent pixels).

public static readonly AdSize BANNER_320x100 (deprecated)

```
public static readonly AdSize BANNER_320x100 = new AdSize {Width = 320, Height = 100, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 320x100 в dp (density-independent pixels).

public static readonly AdSize BANNER_400x240 (deprecated)

```
public static readonly AdSize BANNER_400x240 = new AdSize {Width = 400, Height = 240, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 400x240 в dp (density-independent pixels).

public static readonly AdSize BANNER_728x90 (deprecated)

```
public static readonly AdSize BANNER_728x90 = new AdSize {Width = 728, Height = 90, AdSizeType = AdSizeType.Fixed}
```

Баннер с размерами 728x90 в dp (density-independent pixels).

Класс Banner

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс для показа баннерной рекламы.

Методы

<code>Banner(string blockId, AdSize adSize, AdPosition position)</code>	Создает объект класса Banner с указанными параметрами.
<code>void LoadAd(AdRequest request)</code>	Начинает загрузку рекламы в фоновом режиме.
<code>void Hide()</code>	Скрывает объект класса Banner на экране.
<code>void Show()</code>	Показывает объект класса Banner на экране.
<code>void Destroy()</code>	Уничтожает объект класса Banner и очищает все занятые ресурсы.

События

<code>event EventHandler<EventArgs> OnAdLoaded</code>	Вызывается, когда реклама загружена.
<code>event EventHandler<AdFailureEventArgs> OnAdFailedToLoad</code>	Вызывается, если во время загрузки произошла ошибка.
<code>event EventHandler<EventArgs> OnReturnedToApplication</code>	Вызывается, когда пользователь вернулся в приложение после клика.
<code>event EventHandler<EventArgs> OnLeftApplication</code>	Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).
<code>event EventHandler<EventArgs> OnAdClicked</code>	Вызывается, когда пользователь кликнул на рекламу.
<code>event EventHandler<ImpressionData> OnImpression</code>	Вызывается, когда был засчитан рекламный показ.

Описание методов

Banner(string blockId, AdSize adSize, AdPosition position)

```
public Banner(string blockId, AdSize adSize, AdPosition position)
```

Создает объект класса Banner с указанными параметрами.

Параметры:

blockId	Уникальный идентификатор рекламного места, задается в Партнерском интерфейсе и имеет вид: demo-banner-yandex.
adSize	Размеры баннерной рекламы из класса AdSize.
position	Позиция баннера на экране.

Возвращает:

Объект класса Banner с заданными параметрами.

void LoadAd(AdRequest request)

```
public void LoadAd(AdRequest request)
```

Начинает загрузку рекламы в фоновом режиме.

Параметры:

request	Объект класса AdRequest с данными для таргетированного запроса рекламы.
---------	---

void Hide()

```
public void Hide()
```

Скрывает объект класса Banner на экране.

void Show()

```
public void Show()
```

Показывает объект класса Banner на экране.

void Destroy()

```
public void Destroy()
```

Уничтожает объект класса Banner и очищает все занятые ресурсы.

Описание событий

event EventHandler<EventArgs> OnAdLoaded

```
public event EventHandler<EventArgs> OnAdLoaded
```

Вызывается, когда реклама загружена.

event EventHandler<AdFailureEventArgs> OnAdFailedToLoad

```
public event EventHandler<AdFailureEventArgs> OnAdFailedToLoad
```

Вызывается, если во время загрузки произошла ошибка.

event EventHandler<EventArgs> OnReturnedToApplication

```
public event EventHandler<EventArgs> OnReturnedToApplication
```

Вызывается, когда пользователь вернулся в приложение после клика.

event EventHandler<EventArgs> OnLeftApplication

```
public event EventHandler<EventArgs> OnLeftApplication
```

Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).

event EventHandler<EventArgs> OnAdClicked

```
public event EventHandler<EventArgs> OnAdClicked
```

Вызывается, когда пользователь кликнул на рекламу.

event EventHandler<ImpressionData> OnImpression

```
public event EventHandler<ImpressionData> OnImpression
```

Вызывается, когда был засчитан рекламный показ.

Класс Gender

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Пол пользователя.

Константы

`const string MALE`

Мужской пол.

.

`const string FEMALE`

Женский пол.

.

Описание констант

const string MALE

```
public const string MALE = "male"
```

Мужской пол.

const string FEMALE

```
public const string FEMALE = "female"
```

Женский пол.

Класс Interstitial

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс для показа полноэкранный рекламы.

Методы

<code>Interstitial(string blockId)</code>	Создает объект класса Interstitial с указанными параметрами.
<code>void LoadAd(AdRequest request)</code>	Начинает загрузку рекламы в фоновом режиме.
<code>void IsLoaded()</code>	Возвращает результат загрузки рекламы.
<code>void Show()</code>	Показывает рекламу, если она была загружена.
<code>void Destroy()</code>	Уничтожает объект класса Interstitial и очищает все занятые ресурсы.

События

<code>event EventHandler<EventArgs> OnInterstitialLoaded</code>	Вызывается, когда реклама загружена.
<code>event EventHandler<AdFailureEventArgs> OnInterstitialFailedToLoad</code>	Вызывается, если во время загрузки произошла ошибка.
<code>event EventHandler<EventArgs> OnReturnedToApplication</code>	Вызывается, когда пользователь вернулся в приложение после клика.
<code>event EventHandler<EventArgs> OnLeftApplication</code>	Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).
<code>event EventHandler<EventArgs> OnAdClicked</code>	Вызывается, когда пользователь кликнул на рекламу.
<code>event EventHandler<EventArgs> OnInterstitialShown</code>	Вызывается, когда полноэкранный реклама была показана.
<code>event EventHandler<EventArgs> OnInterstitialDismissed</code>	Вызывается, когда полноэкранный реклама была закрыта.
<code>event EventHandler<ImpressionData> OnImpression</code>	Вызывается, когда был засчитан рекламный показ.
<code>event EventHandler<AdFailureEventArgs> OnInterstitialFailedToShow</code>	Вызывается, когда реклама не может быть показана.

Описание методов

Interstitial(string blockId)

```
public Interstitial(string blockId)
```

Создает объект класса Interstitial с указанными параметрами.

Параметры:

<code>blockId</code>	Уникальный идентификатор рекламного места, задается в Партнерском интерфейсе и имеет вид: demo-banner-yandex.
----------------------	---

Возвращает:

Объект класса Interstitial с заданными параметрами.

void LoadAd(AdRequest request)

```
public void LoadAd(AdRequest request)
```

Начинает загрузку рекламы в фоновом режиме.

Параметры:

request	Объект класса AdRequest с данными для таргетированного запроса рекламы.
---------	---

bool isLoading()

```
public bool isLoading()
```

Возвращает результат загрузки рекламы.

Возвращает:

Значение true — реклама успешно загружена и готова к показу. Если реклама не загружена или не готова к показу, возвращает значение false.

void Show()

```
public void Show()
```

Показывает рекламу, если она была загружена.

void Destroy()

```
public void Destroy()
```

Уничтожает объект класса Interstitial и очищает все занятые ресурсы.

Описание событий

event EventHandler<EventArgs> OnInterstitialLoaded

```
public event EventHandler<EventArgs> OnInterstitialLoaded
```

Вызывается, когда реклама загружена.

event EventHandler<AdFailureEventArgs> OnInterstitialFailedToLoad

```
public event EventHandler<AdFailureEventArgs> OnInterstitialFailedToLoad
```

Вызывается, если во время загрузки произошла ошибка.

event EventHandler<EventArgs> OnReturnedToApplication

```
public event EventHandler<EventArgs> OnReturnedToApplication
```

Вызывается, когда пользователь вернулся в приложение после клика.

event EventHandler<EventArgs> OnLeftApplication

```
public event EventHandler<EventArgs> OnLeftApplication
```

Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).

event EventHandler<EventArgs> OnAdClicked

```
public event EventHandler<EventArgs> OnAdClicked
```

Вызывается, когда пользователь кликнул на рекламу.

event EventHandler<EventArgs> OnInterstitialShown

```
public event EventHandler<EventArgs> OnInterstitialShown
```

Вызывается, когда полноэкранная реклама была показана.

event EventHandler<EventArgs> OnInterstitialDismissed

```
public event EventHandler<EventArgs> OnInterstitialDismissed
```

Вызывается, когда полноэкранная реклама была закрыта.

event EventHandler<ImpressionData> OnImpression

```
public event EventHandler<ImpressionData> OnImpression
```

Вызывается, когда был засчитан рекламный показ.

event EventHandler<AdFailureEventArgs> OnInterstitialFailedToShow

```
public event EventHandler<AdFailureEventArgs> OnInterstitialFailedToShow
```

Вызывается, когда реклама не может быть показана.

Класс Location.Builder

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс для построения пользовательской локации.

Методы

Builder SetLatitude(double latitude)	Задает широту.
Builder SetLongitude(double longitude)	Задает долготу.
Builder SetHorizontalAccuracy(double horizontalAccuracy)	Задает горизонтальную точность.
Location Build()	Создает объект класса Location с данными для пользовательской локации.

Описание методов

Builder SetLatitude(double latitude)

```
public Builder SetLatitude(double latitude)
```

Задает широту.

Параметры:

latitude Широта.

Возвращает:

Объект класса `Location.Builder` с указанной широтой.

Builder SetLongitude(double longitude)

```
public Builder SetLongitude(double longitude)
```

Задаёт долготу.

Параметры:

longitude Долгота.

Возвращает:

Объект класса `Location.Builder` с указанной долготой.

Builder SetHorizontalAccuracy(double horizontalAccuracy)

```
public Builder SetHorizontalAccuracy(double horizontalAccuracy)
```

Задаёт горизонтальную точность.

Параметры:

horizontalAccuracy Горизонтальная точность.

Возвращает:

Объект класса `Location.Builder` с указанной горизонтальной точностью.

Location Build()

```
public Location Build()
```

Создаёт объект класса `Location` с данными для пользовательской локации.

Параметры:

Location Объект класса `Location`.

Возвращает:

Объект класса `Location` с данными для пользовательской локации.

Класс Location

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Пользовательская локация.

Свойства

`double Latitude { get }` Широта.

`double Longitude { get }`

Долгота.

`double HorizontalAccuracy { get }`

Горизонтальная точность.

Описание свойств

`double Latitude { get }`

```
public double Latitude { get }
```

Широта.

`double Longitude { get }`

```
public double Longitude { get }
```

Долгота.

`double HorizontalAccuracy { get }`

```
public double HorizontalAccuracy { get }
```

Горизонтальная точность.

Класс MobileAds



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс содержит общие настройки, которые применяются ко всему Mobile Ads SDK.

Методы

`void SetUserConsent(bool consent)`

Устанавливает значение, которое определяет, разрешил ли пользователь из GDPR-региона сбор персональных данных, используемых для аналитики и таргетирования рекламы. Пользовательские данные не будут собираться до тех пор, пока сбор данных не будет разрешен. Если пользователь однажды разрешил или запретил сбор данных, требуется передавать это значение при каждом запуске приложения.

`void SetLocationConsent(bool consent)`

Определяет, разрешено ли использование локации для загрузки таргетированной рекламы. По умолчанию данные о локации не собираются. Требуется также запросить разрешение на доступ к локации, дополнительно к согласию пользователя на сбор локации.

`void SetAgeRestrictedUser(bool ageRestrictedUser)`

Устанавливает значение, которое определяет является ли пользователь ребенком. По умолчанию считается, что пользователь не является ребенком.

Описание методов

void SetUserConsent(bool consent)

```
public static void SetUserConsent(bool consent)
```

Устанавливает значение, которое определяет, разрешил ли пользователь из GDPR-региона сбор персональных данных, используемых для аналитики и таргетирования рекламы. Пользовательские данные не будут собираться до тех пор, пока сбор данных не будет разрешен. Если пользователь однажды разрешил или запретил сбор данных, требуется передавать это значение при каждом запуске приложения.

Параметры:

consent	Разрешает/запрещает сбор данных. По умолчанию данные не собираются.
---------	---

void SetLocationConsent(bool consent)

```
public static void SetLocationConsent(bool consent)
```

Определяет, разрешено ли использование локации для загрузки таргетированной рекламы. По умолчанию данные о локации не собираются. Требуется также запросить разрешение на доступ к локации, дополнительно к согласию пользователя на сбор локации.

Параметры:

consent	Согласие пользователя на сбор данных о локации. Значение по умолчанию false.
---------	--

void SetAgeRestrictedUser(bool ageRestrictedUser)

```
public static void SetAgeRestrictedUser(bool ageRestrictedUser)
```

Устанавливает значение, которое определяет является ли пользователь ребенком. По умолчанию считается, что пользователь не является ребенком.

Параметры:

ageRestrictedUser	Значение, которое определяет является ли пользователь ребенком. По умолчанию считается, что пользователь не является ребенком.
-------------------	--

Класс RewardedAd



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс для показа рекламы с вознаграждением.

Методы

RewardedAd(string blockId)	Создает объект класса RewardedAd с указанными параметрами.
--	--

void LoadAd(AdRequest request)	Начинает загрузку рекламы в фоновом режиме.
--	---

<code>void IsLoaded()</code>	Возвращает результат загрузки рекламы.
<code>void Show()</code>	Показывает рекламу, если она была загружена.
<code>void Destroy()</code>	Уничтожает объект класса RewardedAd и очищает все занятые ресурсы.

События

<code>event EventHandler<EventArgs> OnRewardedAdLoaded</code>	Вызывается, когда реклама загружена.
<code>event EventHandler<AdFailureEventArgs> OnRewardedAdFailedToLoad</code>	Вызывается, если во время загрузки произошла ошибка.
<code>event EventHandler<EventArgs> OnReturnedToApplication</code>	Вызывается, когда пользователь вернулся в приложение после клика.
<code>event EventHandler<EventArgs> OnLeftApplication</code>	Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).
<code>event EventHandler<EventArgs> OnAdClicked</code>	Вызывается, когда пользователь кликнул на рекламу.
<code>event EventHandler<EventArgs> OnRewardedAdShown</code>	Вызывается, когда реклама была показана.
<code>event EventHandler<EventArgs> OnRewardedAdDismissed</code>	Вызывается, когда реклама была закрыта.
<code>event EventHandler<ImpressionData> OnImpression</code>	Вызывается, когда был засчитан рекламный показ.
<code>event EventHandler<AdFailureEventArgs> OnRewardedAdFailedToShow</code>	Вызывается, когда реклама не может быть показана.
<code>event EventHandler<Reward> OnRewarded</code>	Вызывается, когда пользователь должен быть награжден за просмотр объявления (показ засчитан).

Описание методов

RewardedAd(string blockId)

```
public RewardedAd(string blockId)
```

Создает объект класса RewardedAd с указанными параметрами.

Параметры:

<code>blockId</code>	Уникальный идентификатор рекламного места, задается в Партнерском интерфейсе и имеет вид: <code>demo-banner-yandex</code> .
----------------------	---

Возвращает:

Объект класса RewardedAd с заданными параметрами.

void LoadAd(AdRequest request)

```
public void LoadAd(AdRequest request)
```

Начинает загрузку рекламы в фоновом режиме.

Параметры:

request

Объект класса `AdRequest` с данными для таргетированного запроса рекламы.

bool isLoaded()

```
public bool isLoaded()
```

Возвращает результат загрузки рекламы.

Возвращает:

Значение `true` — реклама успешно загружена и готова к показу. Если реклама не загружена или не готова к показу, возвращает значение `false`.

void Show()

```
public void Show()
```

Показывает рекламу, если она была загружена.

void Destroy()

```
public void Destroy()
```

Уничтожает объект класса `RewardedAd` и очищает все занятые ресурсы.

Описание событий

event EventHandler<EventArgs> OnRewardedAdLoaded

```
public event EventHandler<EventArgs> OnRewardedAdLoaded
```

Вызывается, когда реклама загружена.

event EventHandler<AdFailureEventArgs> OnRewardedAdFailedToLoad

```
public event EventHandler<AdFailureEventArgs> OnRewardedAdFailedToLoad
```

Вызывается, если во время загрузки произошла ошибка.

event EventHandler<EventArgs> OnReturnedToApplication

```
public event EventHandler<EventArgs> OnReturnedToApplication
```

Вызывается, когда пользователь вернулся в приложение после клика.

event EventHandler<EventArgs> OnLeftApplication

```
public event EventHandler<EventArgs> OnLeftApplication
```

Вызывается, когда приложение становится неактивным, так как пользователь кликнул на рекламу и сейчас перейдет в другое приложение (например, браузер).

event EventHandler<EventArgs> OnAdClicked

```
public event event EventHandler<EventArgs> OnAdClicked
```

Вызывается, когда пользователь кликнул на рекламу.

event EventHandler<EventArgs> OnRewardedAdShown

```
public event event EventHandler<EventArgs> OnRewardedAdShown
```

Вызывается, когда реклама была показана.

event EventHandler<EventArgs> OnRewardedAdDismissed

```
public event event EventHandler<EventArgs> OnRewardedAdDismissed
```

Вызывается, когда реклама была закрыта.

event EventHandler<ImpressionData> OnImpression

```
public event event EventHandler<ImpressionData> OnImpression
```

Вызывается, когда был засчитан рекламный показ.

event EventHandler<AdFailureEventArgs> OnRewardedAdFailedToShow

```
public event event EventHandler<IAFailureEventArgs> OnRewardedAdFailedToShow
```

Вызывается, когда реклама не может быть показана.

event EventHandler<Reward> OnRewarded

```
public event event EventHandler<Reward> OnRewarded
```

Вызывается, когда пользователь должен быть награжден за просмотр объявления (показ засчитан).

Класс Reward

**Внимание:**

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс содержит информацию о вознаграждении для пользователя.

Методы

[Reward\(int amount, string type\)](#)

Создает объект класса Reward с указанными параметрами.

Поля

[string amount](#)

Размер награды в каких-либо единицах.

[string type](#)

Тип награды.

Описание методов

Reward(int amount, string type)

```
public Reward(int amount, string type)
```

Создает объект класса Reward с указанными параметрами.

Параметры:

amount	Размер награды в каких-либо единицах.
type	Тип награды.

Описание полей

string amount

```
public readonly string amount
```

Размер награды в каких-либо единицах.

string type

```
public readonly string type
```

Тип награды.

Класс ScreenUtils



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Класс отвечает за преобразование пикселей в density independent пиксели.

Методы

ConvertPixelsToDp(int pixels)	Преобразовывает ширину в пикселях в density independent пиксели.
---	--

Описание методов

ConvertPixelsToDp(int pixels)

```
public static int ConvertPixelsToDp(int pixels)
```

Преобразовывает ширину в пикселях в density independent пиксели.

Параметры:

pixels	Число пикселей.
--------	-----------------

Перечисление AdPosition



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Позиция баннера.

Перечисления

`public enum AdPosition`

Позиция баннера.

Описание перечислений

`public enum AdPosition`

```
public enum AdPosition
{
    TopLeft,
    TopCenter,
    TopRight,
    CenterLeft,
    Center,
    CenterRight,
    BottomLeft,
    BottomCenter,
    BottomRight
}
```

Позиция баннера.

TopLeft

Сверху в левом углу.

TopCenter

Сверху по центру.

TopRight

Сверху в правом углу.

CenterLeft

По центру в левом углу.

Center

По центру.

CenterRight

По центру в правом углу.

BottomLeft

Внизу в левом углу.

BottomCenter

внизу по центру.

BottomRight

Внизу в правом углу.

Класс AdSizeType



Внимание:

Это архивная версия документации. Актуальная документация по всем платформам находится [здесь](#).

Размеры баннера.

Перечисления

`public enum AdSizeType`

Размеры баннера.

Описание перечислений

public enum AdSizeType

```
public enum AdSizeType
{
    Fixed, // deprecated
    Flexible, // deprecated
    Sticky,
    Inline
}
```

Размеры баннера.

Fixed (deprecated)

Баннер с заданной максимальной высотой и шириной.

Flexible (deprecated)

Баннер с заданной максимальной высотой и шириной.

Sticky

Sticky баннер с заданной шириной.

Inline

Встроенный адаптивный баннер с заданной максимальной высотой и шириной.