

## Оглавление

<b>Руководство. Внедрение содержимого Power BI в приложение для клиентов .....</b>	<b>1</b>
Предварительные требования.....	2
Настройка среды разработки для встроенной аналитики .....	3
Регистрация приложения в Azure Active Directory .....	3
Настройка рабочей среды Power BI .....	3
Создать рабочую область .....	3
Создание и публикация отчетов .....	4
Внедрение содержимого с помощью примера приложения .....	5
Идентификатор приложения .....	6
Идентификатор рабочей области .....	7
Идентификатор отчета.....	8
Имя пользователя и пароль Power BI.....	8
Секрет приложения .....	9
Клиент.....	10
Запуск приложения.....	10
Внедрение содержимого в приложении.....	11
Получение элемента содержимого, который нужно внедрить .....	12
Создание маркера внедрения .....	12
Загрузка элемента с помощью JavaScript.....	14
Начало эксплуатации в рабочей среде .....	15
Создание емкости .....	16
Тестирование в ходе разработки .....	17
Назначение рабочей области для емкости.....	17
Дальнейшие действия.....	19

# Руководство. Внедрение содержимого Power BI в приложение для клиентов

- 02.06.2020
- Чтение занимает 11 мин

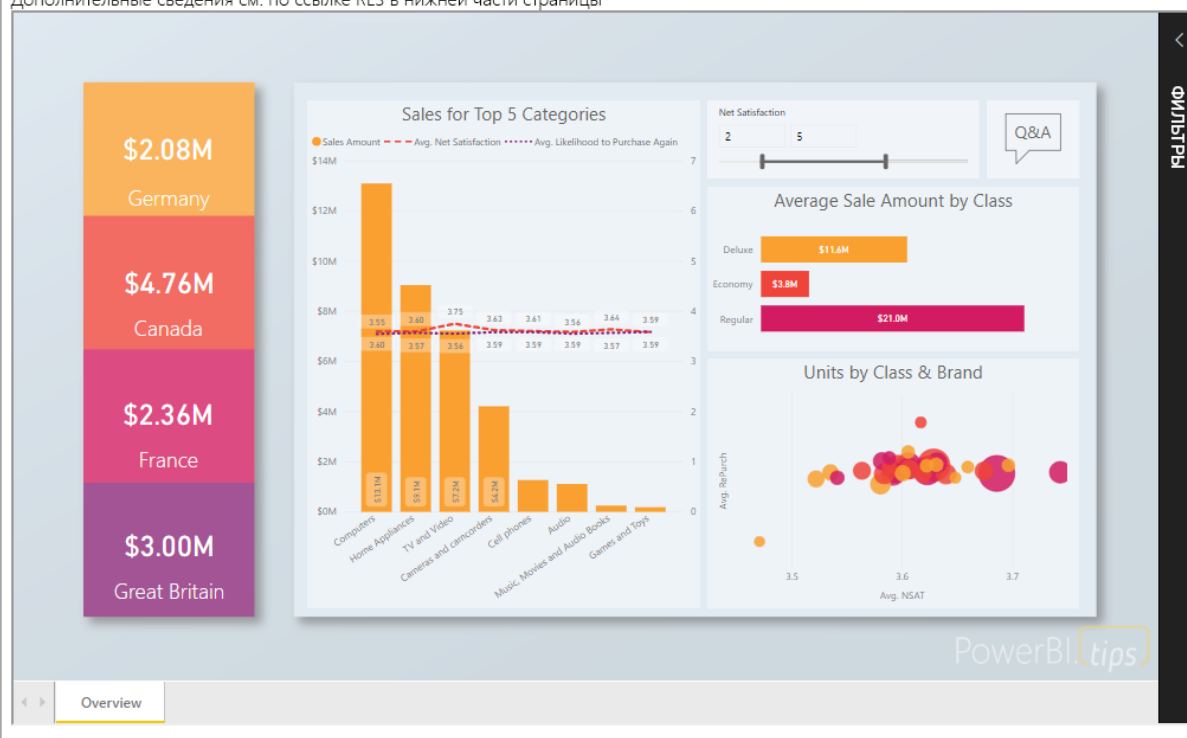
С помощью **Power BI Embedded в Azure** или **внедрения Power BI в Office** вы можете встраивать отчеты, панели мониторинга или плитки в приложение с помощью данных, принадлежащих ему. Модель **данных, принадлежащих приложению**, позволяет приложению использовать Power BI как встроенную платформу аналитики. Как **независимый поставщик программного обеспечения** или **разработчик** вы можете создавать содержимое Power BI, отображающее отчеты, информационные панели или плитки в приложении. При этом приложение будет полностью интегрированным и интерактивным, а пользователям не потребуется лицензия Power BI. Из этого руководства вы узнаете, как интегрировать отчет в приложение с помощью пакета SDK Power BI для .NET и API JavaScript для Power BI.

### Внедренный отчет

Следующий отчет первым найден в заданной группе или имеет значение reportId, определенное в файле Web.config. Вы можете изменить код в соответствии со своими потребностями.

☐ Просмотреть в качестве другого пользователя

Этот флажок недоступен, так как текущий отчет не поддерживает предоставление действующего имени пользователя  
Дополнительные сведения см. по ссылке RLS в нижней части страницы



Из этого руководства вы узнаете, как выполнять следующие задачи:

- регистрация приложения в Azure;
- внедрение отчета Power BI в приложение.

## Предварительные требования

Для работы вам понадобятся:

- [учетная запись Power BI Pro](#) (главная учетная запись, то есть имя пользователя и пароль для входа в учетную запись Power BI Pro) или [субъект-служба \(токен только для приложения\)](#);
- собственная установка [клиента Azure Active Directory](#).

Если вы не зарегистрированы в **Power BI**, перед началом работы [пройдите бесплатную регистрацию](#).

## Настройка среды разработки для встроенной аналитики

Чтобы внедрить в приложение отчеты, панели мониторинга и плитки, необходимо убедиться, что среда допускает внедрение с помощью Power BI.

Воспользуйтесь [средством настройки внедрения](#), чтобы быстро приступить к работе и скачать пример приложения с пошаговой инструкцией для создания среды и внедрения отчета.

Если вы решили настроить среду вручную, см. инструкции ниже.

### Регистрация приложения в Azure Active Directory

[Зарегистрируйте приложение](#) в Azure Active Directory, чтобы предоставить ему доступ к [REST API Power BI](#). Регистрация приложения позволит создать удостоверение для приложения и предоставить ему [разрешения на доступ к ресурсам REST Power BI](#). Способ регистрации приложения зависит от того, используете ли вы главную учетную запись или [субъект-службу](#).

Выбранный метод влияет на тип приложения, регистрируемого в Azure.

Если вы продолжаете использовать главную учетную запись, зарегистрируйте **собственное** приложение, так как в этом случае применяется неинтерактивный вход.

Если же вы используете субъект-службу, необходимо зарегистрировать **веб-приложение на стороне сервера**, чтобы создать секрет приложения.

## Настройка рабочей среды Power BI

### Создать рабочую область

Если вы планируете внедрять отчеты, панели мониторинга и плитки в приложение для клиентов, необходимо разместить содержимое в рабочей области. Можно настроить рабочие области разных типов: [традиционные](#) или [новые](#). Если вы используете *главную* учетную запись, тип рабочих областей может быть

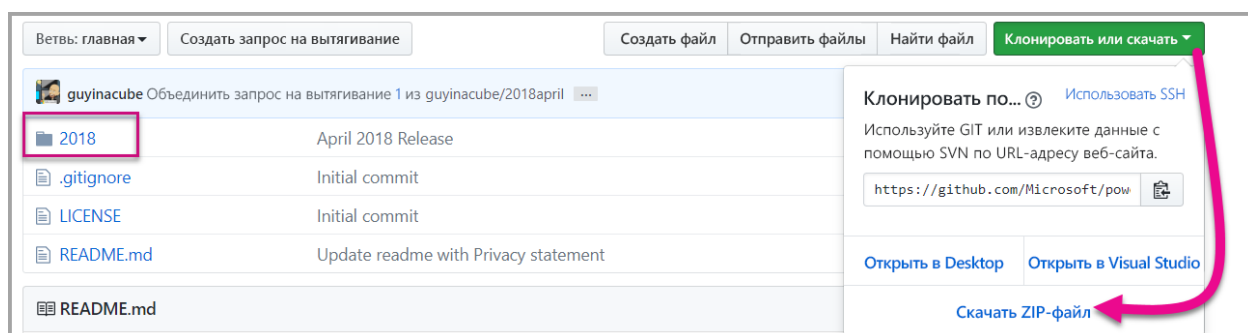
любым. Однако если вы применяете [субъект-службу](#) для входа в приложение, необходимо использовать новые рабочие области. И *главная* учетная запись, и *субъект-служба* должны быть администраторами рабочих областей, в которые входит приложение.

## Создание и публикация отчетов

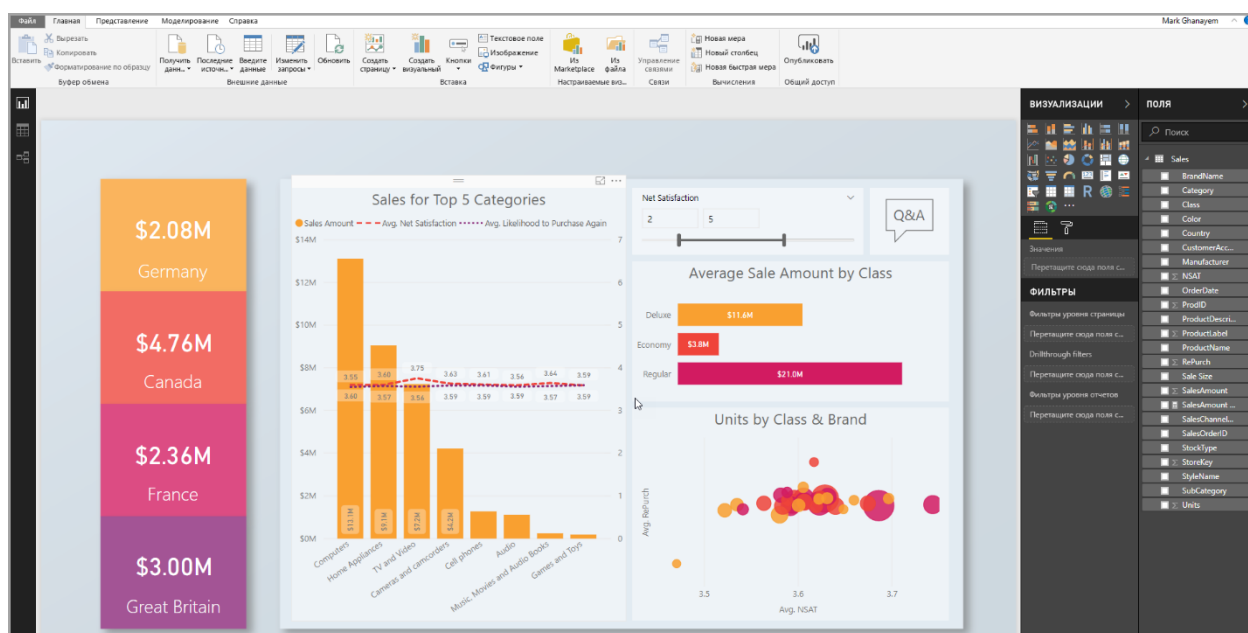
Вы можете создавать отчеты и наборы данных с помощью Power BI Desktop, а затем публиковать эти отчеты в рабочей области. Эту задачу можно выполнять двумя способами. Конечный пользователь может публиковать отчеты в традиционной рабочей области с помощью главной учетной записи (с лицензией Power BI Pro). Если вы используете субъект-службу, то можете публиковать отчеты в новых рабочих областях с помощью [REST API Power BI](#).

Ниже приведены пошаговые инструкции по публикации отчета PBIX в рабочей области Power BI.

1. Скачайте пример [демонстрационной версии блога](#) из GitHub.



2. Откройте PBIX-файл с образцом отчета в **Power BI Desktop**.



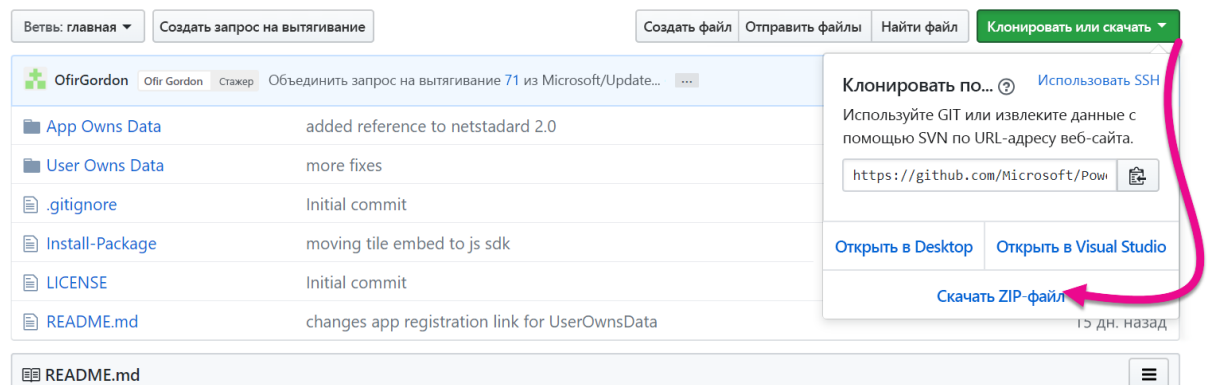
- Опубликуйте его в **рабочих областях** . Этот процесс зависит от того, используете ли вы главную учетную запись (с лицензией Power Pro) или субъект-службу. Если вы используете главную учетную запись, отчет можно опубликовать через Power BI Desktop. Если же вы используете субъект-службу, необходимо применять REST API Power BI.

## Внедрение содержимого с помощью примера приложения

Этот пример намеренно упрощен в целях наглядности. За защиту секрета приложения или учетных данных главной учетной записи отвечаете вы или ваши разработчики.

Чтобы приступить к внедрению содержимого, используя пример приложения, выполните указанные ниже действия.

- Скачайте [Visual Studio](#) (версии 2013 или более поздней). Обязательно скачайте последнюю версию [пакета NuGet](#).
- Чтобы приступить к работе, скачайте [пример с данными, принадлежащими приложению](#) из GitHub.



- Откройте файл **Web.config** из примера приложения. Здесь есть поля, которые нужно заполнить, чтобы успешно запустить приложение. Для поля **AuthenticationType** можно выбрать значение **MasterUser** или **ServicePrincipal** . Выбранный метод проверки подлинности влияет на заполняемые поля.

### Примечание

По умолчанию в этом примере поле **AuthenticationType** имеет значение MasterUser.

ТАБЛИЦА 1

MasterUser (лицензия Power BI Pro)	ServicePrincipal (токен только для приложения)
<a href="#">applicationId</a>	<a href="#">applicationId</a>
<a href="#">workspaceId</a>	<a href="#">workspaceId</a>
<a href="#">reportId</a>	<a href="#">reportId</a>
<a href="#">pbiUsername</a>	
<a href="#">pbiPassword</a>	
	<a href="#">applicationsecret</a>
	<a href="#">tenant</a>

```

Web.config  ▸ ×
25      -->
26      <add key="AuthenticationType" value="MasterUser"/>
27      <!-- Common configuration properties for both authentication types -->
28      <add key="applicationId" value=""/>
29      <add key="workspaceId" value=""/>
30      <!-- The id of the report to embed. If empty, will use the first report in group -->
31      <add key="reportId" value=""/>
32
33      <!-- Fill Tenant ID in authorityUrl-->
34      <add key="authorityUrl" value="https://login.microsoftonline.com/common/" />
35      <add key="resourceUrl" value="https://analysis.windows.net/powerbi/api" />
36      <add key="apiUrl" value="https://api.powerbi.com/" />
37      <add key="embedUrlBase" value="https://app.powerbi.com/" />
38      </appSettings>
39
40      <MasterUser>
41          <!-- Note: Do NOT leave your credentials on code. Save them in secure place. -->
42          <add key="pbiUsername" value=""/>
43          <add key="pbiPassword" value=""/>
44      </MasterUser>
45
46      <ServicePrincipal>
47          <!-- Note: Do NOT leave your app secret on code. Save it in secure place. -->
48          <add key="applicationSecret" value=""/>
49          <add key="tenant" value=""/>
50      </ServicePrincipal>

```

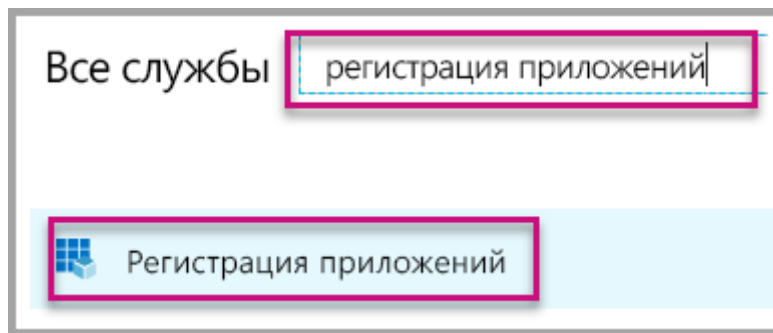
## Идентификатор приложения

Этот атрибут необходим для обоих типов проверки подлинности (главной учетной записи и [субъекта-службы](#)).

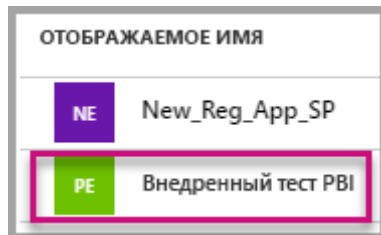
Укажите в поле **applicationId** значение **идентификатора приложения** из **Azure** . Поле **applicationId** используется приложением для его идентификации для пользователей, у которых запрашиваются разрешения.

Чтобы получить значение **applicationId** , сделайте следующее.

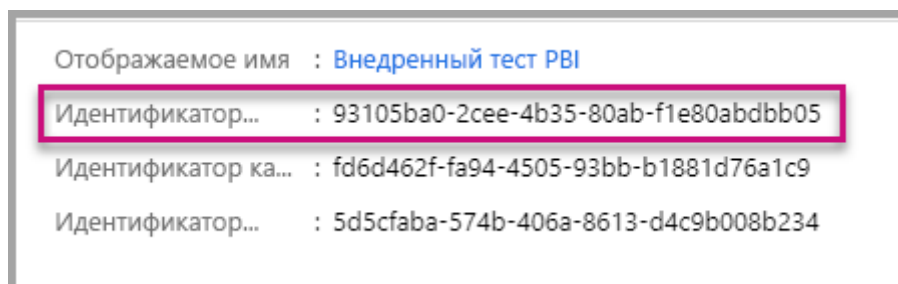
1. Войдите на [портал Azure](#).
2. В области навигации слева выберите **Все службы** , а затем — **Регистрация приложений** .



3. Выберите приложение, для которого требуется **applicationId**.



4. В поле **Идентификатор приложения** указан GUID. Используйте этот **идентификатор приложения** как значение параметра **applicationId** приложения.

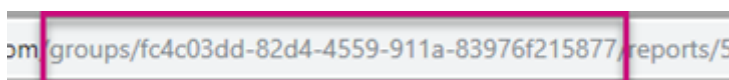


## Идентификатор рабочей области

Этот атрибут необходим для обоих типов проверки подлинности (главной учетной записи и [субъекта-службы](#)).

Укажите в поле **workspaceId** GUID рабочей области (группы) из Power BI. Эти данные можно получить из URL-адреса после входа в службу Power BI или с помощью PowerShell.

URL-адрес



PowerShell

PowerShellКопировать

```
Get-PowerBIworkspace -name "App Owns Embed Test"
```

```
PS C:\> Get-PowerBIworkspace -name "App Owns Embed Test"

Id           : fc4c03dd-82d4-4559-911a-83976f215877
Name         : App Owns Embed test
IsReadOnly   : False
IsOnDedicatedCapacity : True
CapacityId   : 03502BCE-BB57-4768-859A-D28E349F4393
```

## Идентификатор отчета

Этот атрибут необходим для обоих типов проверки подлинности (главной учетной записи и [субъекта-службы](#)).

Укажите в поле **reportId** значение GUID отчета из Power BI. Эти данные можно получить из URL-адреса после входа в службу Power BI или с помощью PowerShell.

URL-адрес

76f215877/reports/5f622ac0-6c2b-45ca-9c72-76fe32ecb303/Reports

PowerShell

PowerShellКопировать

`Get-PowerBIworkspace -name "App Owns Embed Test" | Get-PowerBIReport`

```
PS C:\> Get-PowerBIworkspace -Name 'App Owns Embed Test' | Get-PowerBIReport

Id           : 5f622ac0-6c2b-45ca-9c72-76fe32ecb303
Name         : 2018SU06 Blog Demo - June
WebUrl       : https://msit.powerbi.com/groups/fc4c03dd-82d4-4559-911a-83976f215877-76fe32ecb303
EmbedUrl     : https://msit.powerbi.com/reportEmbed?reportId=5f622ac0-6c2b-45ca-d-82d4-4559-911a-83976f215877&config=eyJjbHVzdGVyVXJsIjoiaHR0cHM6W5hbHlzaXMud2luZG93cy5uZXQifQ%3d%3d
DatasetId    : 4b043c3c-dfdb-45bf-9878-ba6a2150f30c
```

## Имя пользователя и пароль Power BI

Эти атрибуты необходимы, только если в качестве типа проверки подлинности выбрана главная учетная запись.

Если для проверки подлинности используется [субъект-служба](#), заполнять атрибуты имени пользователя и пароля не нужно.



- Укажите в поле **pbiUsername** главную учетную запись Power BI.
- Укажите в поле **pbiPassword** пароль для главной учетной записи Power BI.

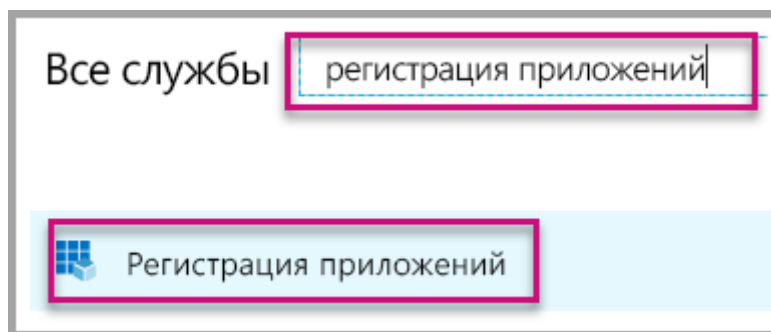
## Секрет приложения

Этот атрибут необходим только для типа проверки подлинности [субъект-служба](#).

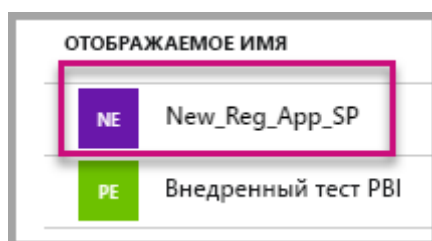
Заполните сведения **ApplicationID** из раздела **Ключи** раздела **Регистрация приложений** в **Azure** . Этот атрибут действует при использовании [субъекта-службы](#).

Чтобы получить значение **ApplicationSecret** , сделайте следующее.

1. Войдите на [портал Azure](#).
2. В области навигации слева выберите **Все службы** , а затем — **Регистрация приложений** .



3. Выберите приложение, для которого нужно использовать **ApplicationSecret** .



4. Выберите **Сертификаты и секреты** в разделе **Управление** .
5. Выберите **Новые секреты клиента** .
6. Введите имя в поле **Описание** и выберите длительность. Затем выберите **Сохранить** , чтобы получить **Значение** для вашего приложения. Когда вы закроете панель **Ключи** после сохранения значения ключа, поле значения отображается только как скрытое. На этом этапе вы не можете получить значение ключа. Если вы потеряете значение ключа, потребуется создать новое на портале Azure.

<b>Секреты клиента</b> Строка секрета, которую приложение использует для подтверждения идентификации при запросе токена. Также он иногда называется паролем приложения.		
<a href="#">+ Новый секрет клиента</a>		
ОПИСАНИЕ	СРОК ДЕЙСТВИЯ	ЗНАЧЕНИЕ
Тест PBI	31.12.2299	xKWB)L5A_Lug-2A1=Vb.2

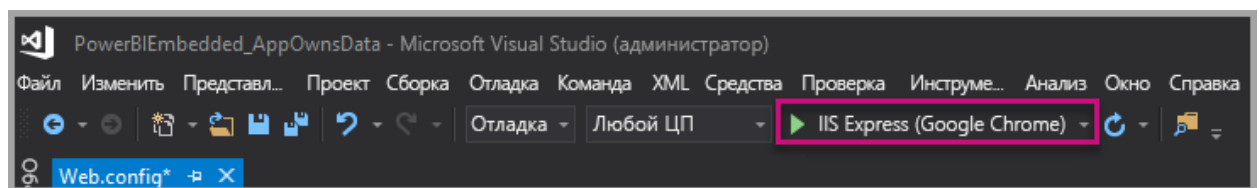
## Клиент

Этот атрибут необходим только для типа проверки подлинности [субъект-служба](#).

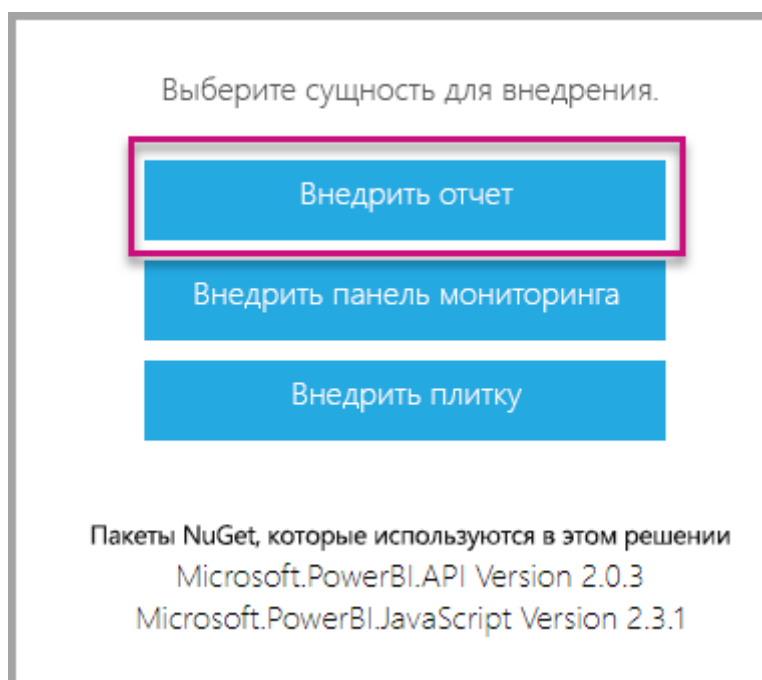
Укажите в поле **tenant** идентификатор клиента Azure. Эти данные можно получить в [центре администрирования Azure AD](#) после входа в службу Power BI или с помощью PowerShell.

## Запуск приложения

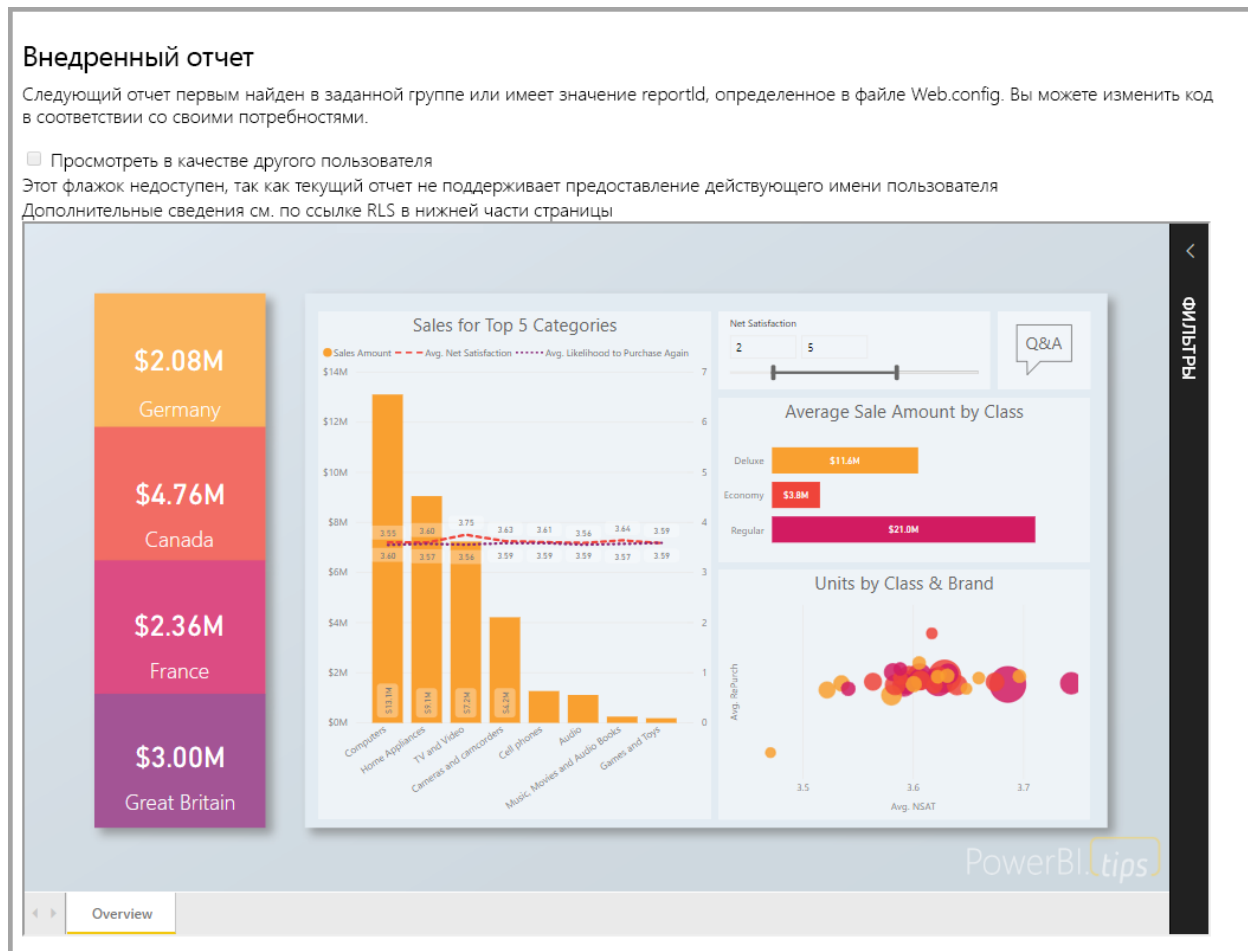
1. Выберите **Запуск** в **Visual Studio** .



2. Теперь выберите **Внедрить отчет** . В зависимости от того, какое содержимое вы хотите проверить (отчеты, панели мониторинга или плитки), выберите нужный вариант в приложении.



### 3. Теперь вы можете просмотреть отчет в примере приложения.



## Внедрение содержимого в приложении

Несмотря на то, что шаги по внедрению содержимого выполняются с помощью [интерфейсов REST API Power BI](#), примеры кода, описанные в этой статье, созданы с помощью **пакета SDK для .NET**.

Чтобы внедрять содержимое для клиентов в приложении, вам нужно получить **маркер доступа** для главной учетной записи или [субъекта-службы](#) из **Azure AD**. [Маркер доступа Azure AD](#) для приложения Power BI необходимо получить, прежде чем выполнять вызовы [REST API Power BI](#).

Для создания клиента Power BI с помощью **маркера доступа** потребуется создать объект клиента Power BI, что позволит работать с интерфейсами [REST API Power BI](#). Для создания объекта клиента Power BI маркер **AccessToken** упаковывается в объект **Microsoft.Rest.TokenCredentials**.

C#Копировать

```
using Microsoft.IdentityModel.Clients.ActiveDirectory;  
using Microsoft.Rest;  
using Microsoft.PowerBI.Api.V2;
```

```
var tokenCredentials = new TokenCredentials(authenticationResult.AccessToken,
"Bearer");

// Create a Power BI Client object. it's used to call Power BI APIs.
using (var client = new PowerBIClient(new Uri(ApiUrl), tokenCredentials))
{
    // Your code to embed items.
}
```

## Получение элемента содержимого, который нужно внедрить

Вы можете использовать объект клиента Power BI, чтобы получить ссылку на элемент, который требуется внедрить.

Ниже приведен пример кода для получения первого отчета из определенной рабочей области.

*Пример получения элемента содержимого (отчета, панели мониторинга или плитки), который вы хотите внедрить, можно найти в файле `Services\EmbedService.cs` в [примере приложения](#).*

```
C#Копировать
using Microsoft.PowerBI.Api.V2;
using Microsoft.PowerBI.Api.V2.Models;

// You need to provide the workspaceId where the dashboard resides.
ODataResponseListReport reports = await
client.Reports.GetReportsInGroupAsync(workspaceId);

// Get the first report in the group.
Report report = reports.Value.FirstOrDefault();
```

## Создание маркера внедрения

Создайте токен внедрения, который можно использовать из API JavaScript. Существует два типа API; первая группа содержит пять API, каждый из которых создает токен внедрения для определенного элемента. Вторая группа, которая содержит только один API, создает токен, который можно использовать для внедрения нескольких элементов.

## Интерфейсы API для создания токена внедрения для определенного элемента

Токен внедрения, созданный этими API, будет связан только с внедряемым элементом. При каждом внедрении элемента Power BI (такого как отчет, панель мониторинга или плитка) при помощи этих API необходимо создать для него новый токен внедрения.

- [Панели мониторинга: GenerateTokenInGroup](#)
- [Наборы данных: GenerateTokenInGroup](#)
- [Отчеты: GenerateTokenForCreateInGroup](#)
- [Отчеты: GenerateTokenInGroup](#)
- [Плитки: GenerateTokenInGroup](#)

Примеры создания токена внедрения для отчета, панели мониторинга или плитки доступны в следующих файлах [примеров приложений](#).

- Services\EmbedService.cs
- Models\EmbedConfig.cs
- Models\TileEmbedConfig.cs

Ниже приведен пример кода для использования API при создании токена внедрения отчетов GenerateTokenInGroup.

C#Копировать

```
using Microsoft.PowerBI.Api.V2;
using Microsoft.PowerBI.Api.V2.Models;

// Generate Embed Token.
var generateTokenRequestParameters = new GenerateTokenRequest(accessLevel: "view");
EmbedToken tokenResponse = client.Reports.GenerateTokenInGroup(workspaceId,
report.Id, generateTokenRequestParameters);

// Generate Embed Configuration.
var embedConfig = new EmbedConfig()
{
    EmbedToken = tokenResponse,
    EmbedUrl = report.EmbedUrl,
    Id = report.Id
};
```

### **API для создания и внедрения токена для нескольких элементов**

API [создания токена](#) внедрения создает токен, который можно использовать для внедрения нескольких элементов.

Он также допускает динамический выбор набора данных при внедрении отчета. Дополнительные сведения о таком использовании API см. в разделе [Динамическая привязка](#).

Ниже приведен пример использования этого API.

C#Копировать

```
using Microsoft.PowerBI.Api.V2;
using Microsoft.PowerBI.Api.V2.Models;
```

```

var reports = new List<GenerateTokenRequestV2Report>()
{
    new GenerateTokenRequestV2Report()
    {
        AllowEdit = false,
        Id = report1.Id
    },
    new GenerateTokenRequestV2Report()
    {
        AllowEdit = true,
        Id = report2.Id
    }
};

var datasets= new List<GenerateTokenRequestV2Dataset>()
{
    new GenerateTokenRequestV2Dataset(dataset1.Id),
    new GenerateTokenRequestV2Dataset(dataset2.Id),
    new GenerateTokenRequestV2Dataset(dataset3.Id),
};

var targetWorkspaces = new List<GenerateTokenRequestV2TargetWorkspace>()
{
    new GenerateTokenRequestV2TargetWorkspace(workspace1.Id),
    new GenerateTokenRequestV2TargetWorkspace(workspace2.Id),
};

var request = new GenerateTokenRequestV2()
{
    Datasets = datasetsRequestDetails ?? null,
    Reports = reportsRequestDetails,
    TargetWorkspaces = targetWSRequestdetials ?? null,
};

var token = client.GetClient().EmbedToken.GenerateToken(request);

```

## Загрузка элемента с помощью JavaScript

Чтобы загрузить отчет в элемент div веб-страницы, вы можете использовать JavaScript.

С полнофункциональным примером на основе API JavaScript вы можете ознакомиться на странице [средства "Площадка"](#). Средство "Тестовая площадка" — это очень удобный способ быстро протестировать разные примеры Power BI Embedded. Дополнительные сведения об API JavaScript вы найдете на странице [вики-сайта PowerBI-JavaScript](#).

В примере используются модели **EmbedConfig** и **TileEmbedConfig** и представления отчета.

Пример добавления представления для отчета, панели мониторинга или плитки доступен в файлах Views\Home\EmbedReport.cshtml, Views\Home\EmbedDashboard.cshtml или Views\Home\Embedtile.cshtml в [примере приложения](#).

JavaScriptКопировать

```
<script src="~/scripts/powerbi.js"></script>
<div id="reportContainer"></div>
<script>
    // Read embed application token from Model
    var accessToken = "@Model.EmbedToken.Token";

    // Read embed URL from Model
    var embedUrl = "@Html.Raw(Model.EmbedUrl)";

    // Read report Id from Model
    var embedReportId = "@Model.Id";

    // Get models. models contains enums that can be used.
    var models = window['powerbi-client'].models;

    // Embed configuration used to describe what and how to embed.
    // This object is used when calling powerbi.embed.
    // This also includes settings and options such as filters.
    // You can find more information at https://github.com/Microsoft/PowerBI-
JavaScript/wiki/Embed-Configuration-Details.
    var config = {
        type: 'report',
        tokenType: models.TokenType.Embed,
        accessToken: accessToken,
        embedUrl: embedUrl,
        id: embedReportId,
        permissions: models.Permissions.All,
        settings: {
            filterPaneEnabled: true,
            navContentPaneEnabled: true
        }
    };

    // Get a reference to the embedded report HTML element
    var reportContainer = $('#reportContainer')[0];

    // Embed the report and display it within the div container.
    var report = powerbi.embed(reportContainer, config);
</script>
```

## Начало эксплуатации в рабочей среде

Завершив разработку приложения, вернитесь к рабочей области с емкостью.

## Важно!

Для перехода к рабочей среде требуется емкость. Все рабочие области, содержащие отчеты и панели мониторинга, а также содержащие наборы данных, должны быть назначены емкости.

## Создание емкости

Создав емкость, вы сможете пользоваться ресурсом для клиента. Можно использовать два типа емкости.

- **Power BI Premium** — это подписка Office 365 уровня клиента, доступная в двух семействах SKU: *EM* и *P*. При внедрении содержимого Power BI это решение называется *внедрением Power BI*. Дополнительные сведения об этой подписке см. в разделе [Что такое Power BI Premium?](#).
- **Azure Power BI Embedded** — вы можете приобрести емкость на [портале Microsoft Azure](#). Эта подписка использует номера SKU *A*. Дополнительные сведения о создании емкости Power BI Embedded см. в статье [Создание емкости Power BI Embedded на портале Azure](#).

## Примечание

С номерами SKU *A* вы не можете получить доступ к содержимому Power BI с бесплатной лицензией Power BI.

В следующей таблице описаны ресурсы и ограничения для каждого SKU. Чтобы определить, какая емкость лучше соответствует вашим потребностям, ознакомьтесь с таблицей [Какой номер SKU следует приобрести для моего сценария](#).

Узлы емкости	Число виртуальных ядер	Серверные виртуальные ядра	ОЗУ (ГБ)	Интерфейсные виртуальные ядра	Подключения DirectQuery и активные подключения (в секунду)
EM1/A1	1	0,5	2.5	0,5	3,75 %
EM2/A2	2	1	5	1	7,5 %
EM3/A3	4	2	10	2	15
P1/A4	8	4	25	4	30
P2/A5	16	8	50	8	60
P3/A6	32	16	100	16	120



## Тестирование в ходе разработки

Для тестирования в процессе разработки можно использовать токены внедрения пробной версии с лицензией Pro. Для внедрения в рабочей среде используйте емкость.

Число токенов внедрения пробной версии, которые можно создать в субъекте-службе или основной учетной записи Power BI, ограничено. Используйте API [доступных компонентов](#), чтобы просмотреть текущий объем использования сценариев внедрения в процентном соотношении. Объем использования приводится для отдельного субъекта-службы или главной учетной записи.

Если во время тестирования токены внедрения закончились, необходимо приобрести [емкость](#) Power BI Embedded или Premium. В этом случае не будет ограничения на количество создаваемых токенов.

## Назначение рабочей области для емкости

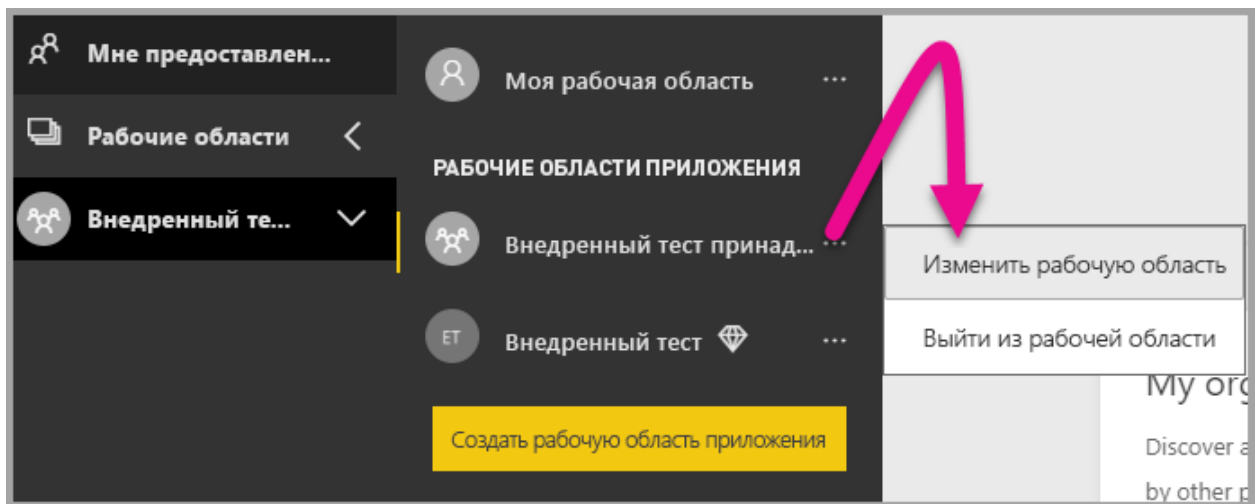
Созданной емкости можно назначить рабочую область.

Все рабочие области, которые содержат ресурсы Power BI, связанные с внедренным содержимым (включая наборы данных, отчеты и панели мониторинга), должны быть назначены емкостям. Например, если внедренный отчет и связанный с ним набор данных размещаются в разных рабочих областях, обе эти рабочие области должны быть назначены емкостям.

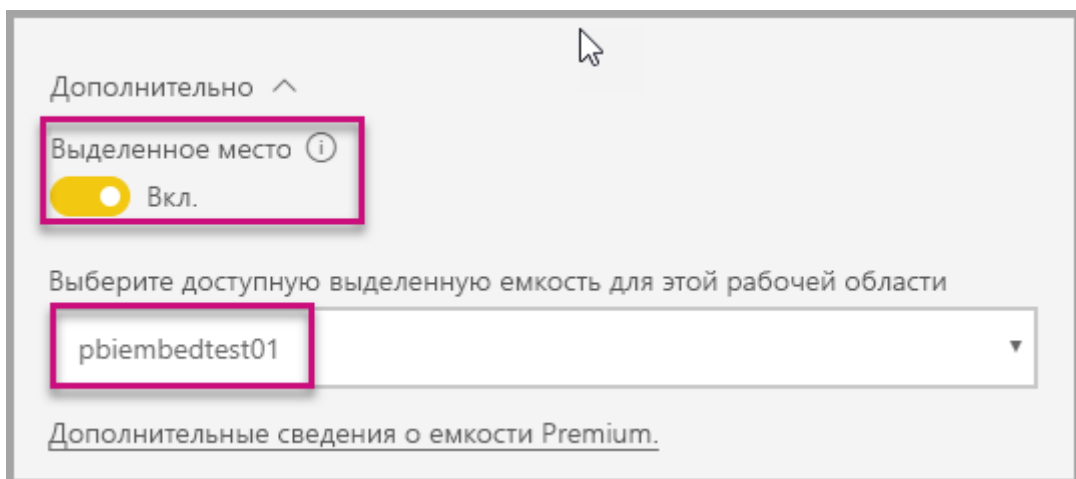
Чтобы назначить емкость рабочей области с помощью [субъекта-службы](#), используйте [REST API Power BI](#). Если применяется REST API Power BI, необходимо использовать [идентификатор объекта субъекта-службы](#).

Чтобы назначить емкость рабочей области с помощью **главной учетной записи**, выполните указанные ниже действия.

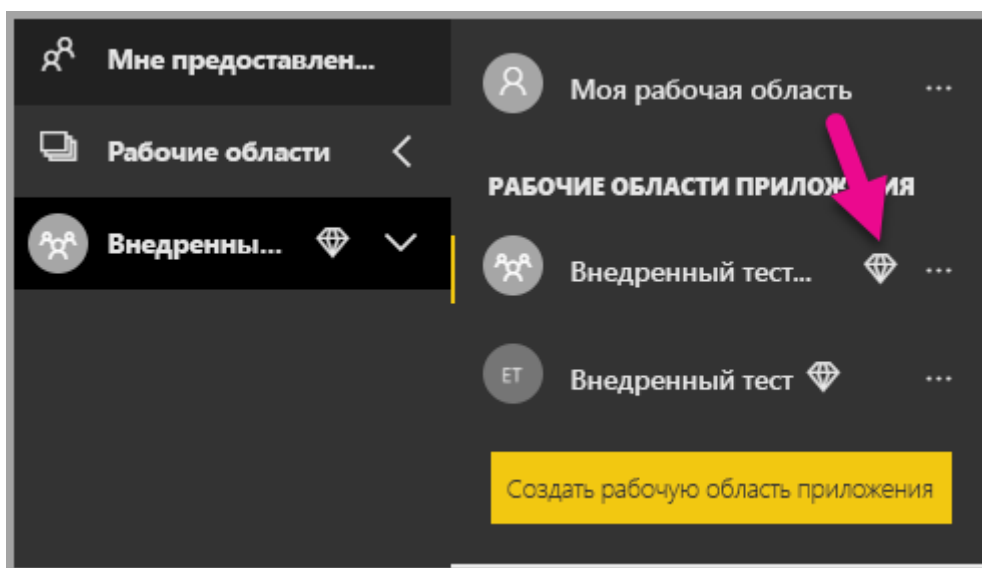
1. В **службе Power BI** разверните рабочие области и щелкните многоточие возле рабочей области, которую вы используете для внедрения содержимого. Затем выберите команду **Изменить рабочие области**.



2. Разверните элемент **Дополнительно** , затем включите параметр **Емкость** и выберите созданную емкость. Затем выберите **Сохранить** .



3. После нажатия кнопки **Сохранить** рядом с именем рабочей области должен появиться **ромб** .



## Дальнейшие действия

В этом руководстве вы узнали, как внедрить содержимое Power BI в свое приложение для клиентов. Кроме того, вы можете попробовать внедрить содержимое Power BI для своей организации.

[Внедрение для организации](#)