

# HELPPAGE



# FAVI EXTRA PLUGIN PRO GTM



## Instalace FAVI Partner Events Tracking

Celou implementaci máte do 15 minut hotovou.

Pro ty, kteří s GTM nemají žádné zkušenosti je návod podrobnější a doporučujeme aby si manuál přečetli celý.

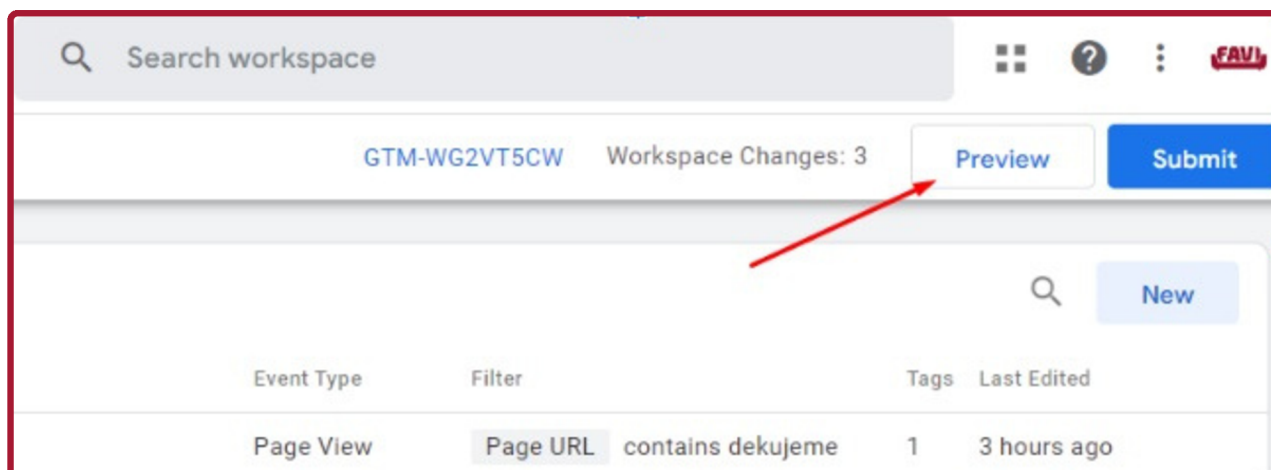
Samozřejmě tu jsme také od toho, abychom vám s čímkoliv pomohli. Proto neváhejte posílat své dotazy na [lukas.homola@favi.cz](mailto:lukas.homola@favi.cz) a rovnou sdílet svůj GTM účet na tento e-mail [partner-events-tracking-support@favionline.com](mailto:partner-events-tracking-support@favionline.com) pro usnadnění a urychlení spolupráce.

## Zjišťování vašich parametrů z objednávky pomocí GTM debugger

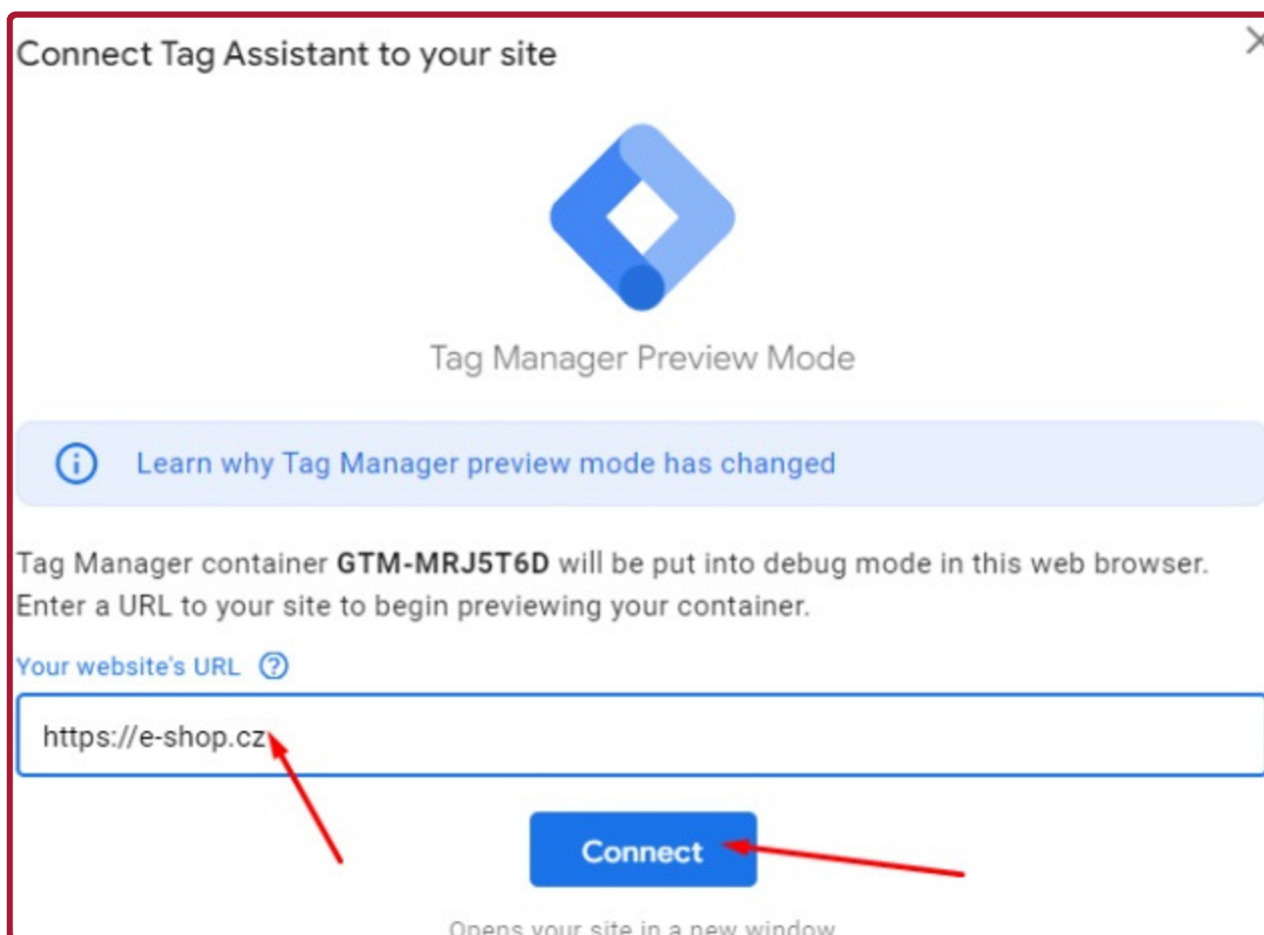
Doporučujeme si stáhnout tento doplněk do prohlížeče “Tag Assistant Legacy” (bez tohoto doplňku nebudete schopni dobře přečíst data z vašeho webu):

- [Google Chrome](#)
- [Mozilla Firefox](#)
- [Opera](#)
- [Edge](#)

1. Ve vašem účtu klikněte na “Náhled” v horní části



2. Zkopírujte URL vašeho e-shopu a klikněte na “Připojit”



3. Nyní udělejte testovací objednávku až do fáze, kdy je produkt úspěšně kompletně objednan

4. Najděte si ten spouštěč u kterého se propsala objednávka produktu. V našem případě se jedná o event "Purchase" a otevřete si nabídku "DataLayer"

- Správný spouštěč poznáte tak, že v nabídce DataLayer budete mít výpis informací o produktu a prodávajícím (viz obrázek)

The screenshot displays the Google Tag Manager interface for an event named 'purchase'. On the left, a list of triggers is shown, with '80 purchase' selected. The main area shows the event configuration, including the API Call: `dataLayer.push({event: "purchase", ...})`. Below this, the 'Output of GTM-P5JZFZ3H' is shown, with the 'Data Layer' tab selected. The Data Layer values are displayed as a JSON object:

```
1 {
2   event: "purchase",
3   gtm: {uniqueEventId: 33, start: 1702311886912},
4   eventModel: {
5     value: "1220",
6     currency: "CZK",
7     transaction_id: "6031657",
8     items: [
9       {
10        item_id: "4972_108023",
11        item_name: "Jídelní židle BOS 4 D, dřevo: bílá",
12        price: 1220,
13        quantity: "1"
14      }
15    ]
16  }
17 }
```

5. Zde nás (v našem případě) zajímají tyto řádky:

- “**eventModel**” - Nebo i jiný parametr, kde mám vypsané transakční hodnoty o produktu
- “**items**” - zde mám vypsané informace o produktu, jako jsou například:
  - “**item\_id**” - ID produktu
  - “**item\_name**” - Název produktu
  - U vás to může být opět rozdílné, ale jde o výše vyjmenované hodnoty

Tags Variables **Data Layer**

Data Layer values after this Message:

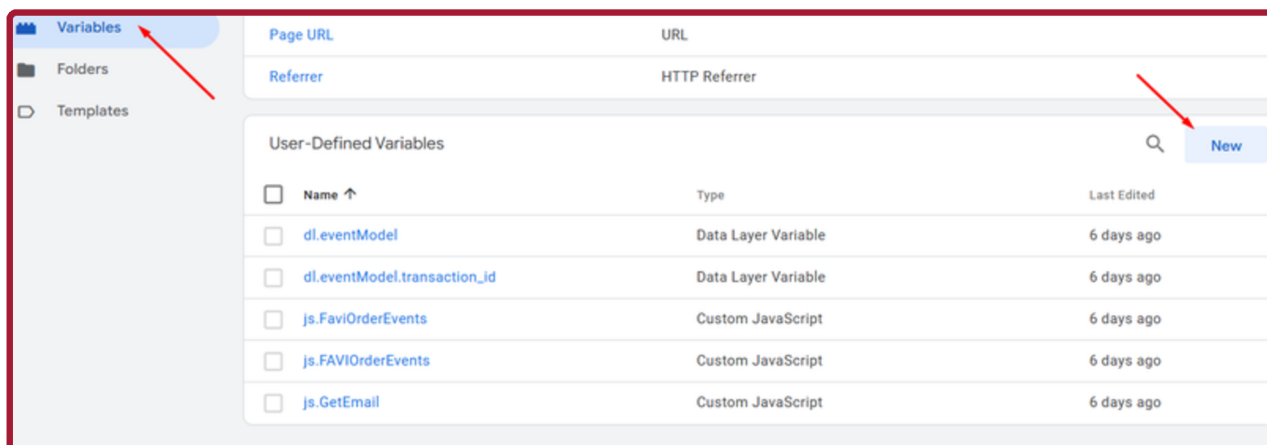
```
1 {
2   event: "purchase",
3   gtm: {uniqueEventId: 33, start: 1702311886912},
4   eventModel: { ←
5     value: "1220",
6     currency: "CZK",
7     transaction_id: "6031657",
8     items: [
9       {
10        ← item_id: "4972_108023",
11        ← item_name: "Jídelní židle BOS 4 D, dřevo: bílá",
12        price: 1220,
13        quantity: "1"
14      }
15    ]
16  }
17 }
```

8. Všechny tyto hodnoty (co jsou výše označené) nyní vložíme do vlastní proměnné v GTM

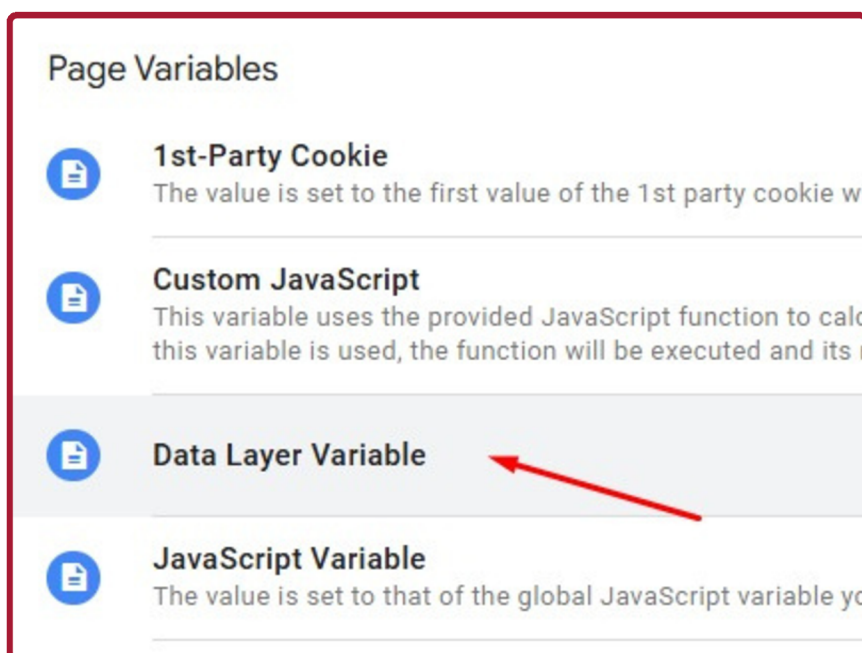
## Zjišťování vašich parametrů z objednávky pomocí GTM debugger

1. V první řadě si vytvoříme proměnné z Data Layer vrstvy pro hodnoty transakcí z dané objednávky

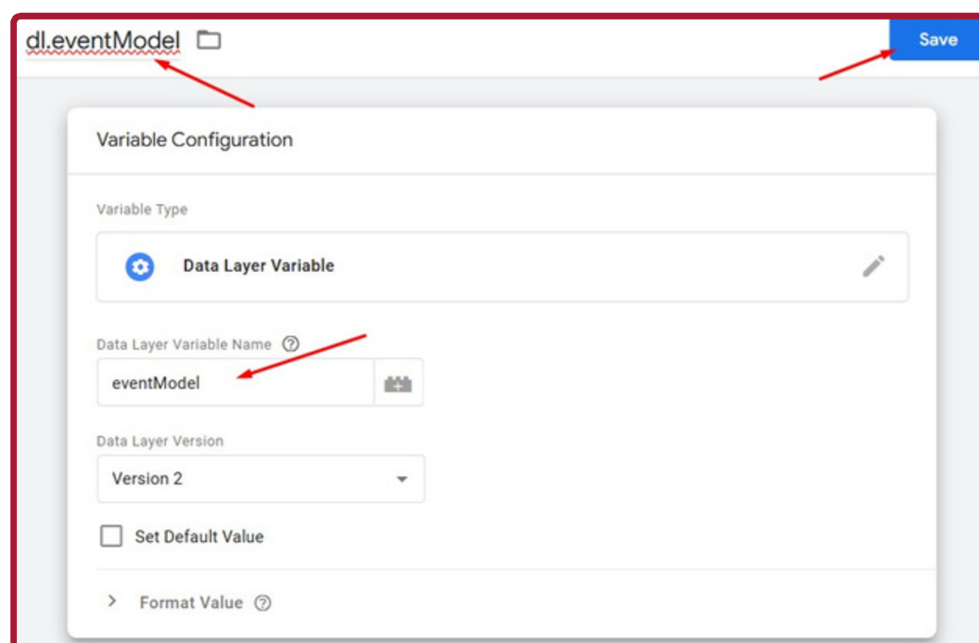
- Vyberte z levého sloupce nabídku “Proměnné” a vyberte “Nový”
- V pravém sloupci si najděte “Proměnná Datové Vrstvy” a vyberte ji
- Do prázdného pole zadáte parametr pod kterým vidíte objednávku z vaší datové vrstvy (v našem případě se jedná o “**eventModel**”)
- A pojmenujte DL proměnnou rovnou “**dl.eventModel**” a uložte



Zde vyberte z proměnných “Data Layer Variable”



Do pole “Data Layer Variable Name” zadáte přesný název parametru z vaší datové vrstvy a uložíte ji pod názvem dl.XXX



2. Nyní celou akci opakujte a vytvoříme nyní proměnou z DL vrstvy pro hodnoty dané objednávky

- Do prázdného pole nyní napište “eventModel.transaction\_id”
- Tečka mezi parametry nám značí, že parametr “transaction\_id” je zanořený v parametru “eventModel” (je to tzv. cesta)
- **POZOR: Vždy uchovávejte naprosto přesné názvy parametru, jako máte na svém webu!!**



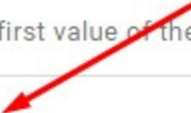


3. A naposledy opakujte akci pro vytvoření poslední proměnné pro DL a nyní to bude vytvoření parametru pro získání e-mailu zákazníka

- V našem případě máme vytvořenou proměnnou s názvem “dl.custom.email”

4. V tuto chvíli máme vytvořené podstatné proměnné z DL a vytvoříme poslední část proměnné pro “Custom JavaScript”

- Vyberte “nový” v proměnných a vyberte “Vlastní proměnná JavaScript”

### Page Variables

-  **1st-Party Cookie**  
The value is set to the first value of the 1st party cookie with the matching name.
-  **Custom JavaScript**   
This variable uses the provided JavaScript function to calculate its value in the browser. Each time this variable is used, the function will be executed and its return value will be used.
-  **Data Layer Variable**
-  **JavaScript Variable**  
The value is set to that of the global JavaScript variable you specify.



5. Zde vyplňte název proměnné například “**js.FaviOrderEvents**” a můžeme se pustit do vložení vlastního kódu



6. Nyní je zapotřebí vložit Function Code s proměnnými a správně je namapovat.

Samozřejmě jsou názvy proměnných na vás, ale nezapomeňte si je změnit v níže přiloženém kódu, aby nenastala chyba.

7. Níže kód je již upravený, dle tohoto návodu, takže stačí jen kopírovat a vložit

- V případě problému níže přiloženého kódu, doporučujeme jej zkopírovat z této URL (ALE, kód je v základní verzi bez úpravy) <https://github.com/favionline/partner-events-tracking-gtm-template/blob/master/doc/implementation-guide/readme.md>

```

function() {
  var order = {{dl.eventModel}};

  return {
    orderId: order.transaction_id.toString(),
    orderItems: order.items.map(function (item) {
      return {
        product: {
          id: item.item_id.toString(),
          name: item.item_name,
        },
      };
    }),
    customer: {
      email: {{dl.custom.email}},
    },
  };
}

```

Celý výstup vašeho kódu může pak vypadat takto

```

1 function() {
2   var order = {{dl.eventModel}}; ← Proměnná z Data Layer
3
4   return {
5     orderId: order.transaction_id.toString(), ← Namapování z DataLayer
6     orderItems: order.items.map(function (item) {
7       return {
8         product: {
9           id: item.item_id.toString(),
10          name: item.item_name,
11        },
12      };
13    }),
14    customer: {
15      email: {{js.custom.email}}, ← Proměnná z DataLayer
16    },
17  };
18 }

```

## 8. Proměnná v GTM se vždy značí těmito znaky `{{}}`

- Když začnete řádek s proměnnou prvními dvěma špičatými závorkami “{{”, tak vám GTM už bude vyjíždět nabídky, které můžete vložit jako proměnou
- Samozřejmě musíte mít proměnou už uloženou (viz výše kroky)

## 9. Osmý a devátý řádek je cesta item (položky) k hodnotám ID a názvu produktu

- Zde proto stačí pouze uvést cestu v poli k daným hodnotám
- Viz tento obrázek z testování (poslední dvě šipky)

The screenshot shows the Google Tag Manager interface. On the left, a list of triggers is visible, with '80 purchase' selected. The main area displays the configuration for an 'Event: purchase' tag. The tag type is 'API Call' with the code `dataLayer.push({event: "purchase", ...})`. Below this, the 'Output of GTM-P5JZFZ3H' is shown, with the 'Data Layer' tab selected. The Data Layer values after the message are displayed as a JSON object:

```
1 {
2   event: "purchase",
3   gtm: {uniqueEventId: 33, start: 1702311886912},
4   eventModel: {
5     value: "1220",
6     currency: "CZK",
7     transaction_id: "6031657",
8     items: [
9       {
10      item_id: "4972_108023",
11      item_name: "Jídelní židle BOS 4 D, dřevo: bílá",
12      price: 1220,
13      quantity: "1"
14    }
15  ]
16 }
17 }
```

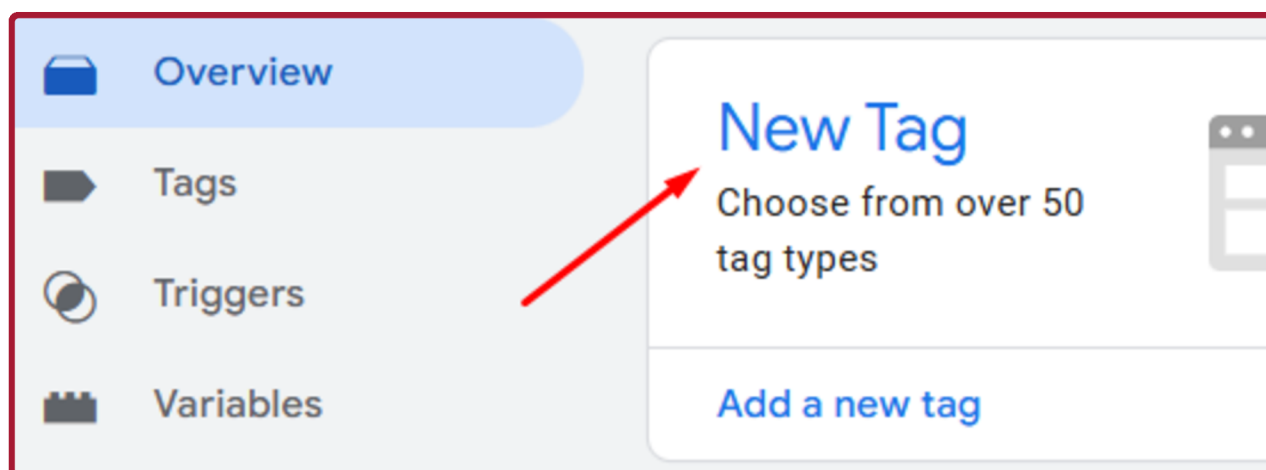
Red arrows point to the `eventModel` property on line 4 and the `items` array on line 8. Two more red arrows point to the `item_id` and `item_name` properties on lines 10 and 11 respectively.

10. Tato proměnná mi nyní posílá hodnoty z hotové objednávky a je vytvořena následujícím způsobem

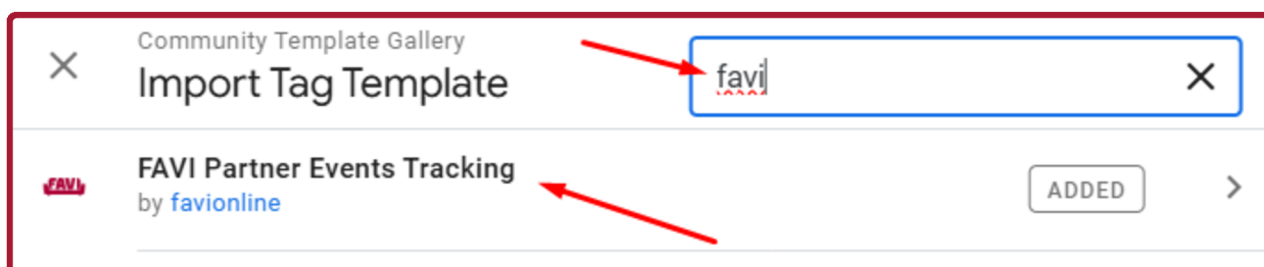
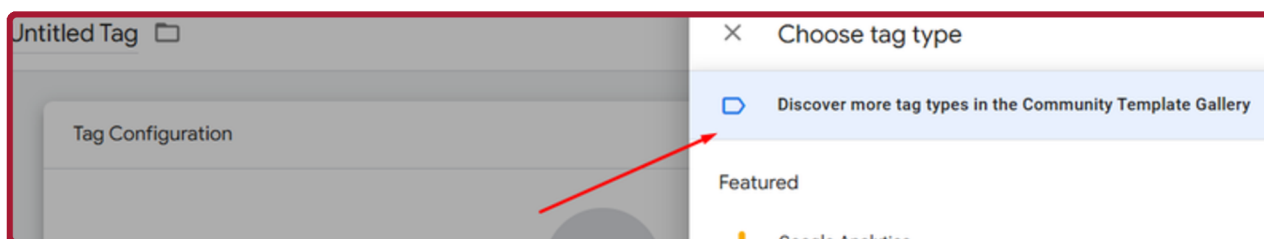
- Function nyní čte:
  - přes proměnou “**dl.eventModel**” - transakční údaje o objednávce
  - přes namapování “**order.transaction\_id**” - ID transakce
  - přes “**item.item\_id**” - Id produktu
  - přes “**item.item\_name**” - název produktu
  - přes “**dl.customer.email**” - E-mail zákazníka, který právě produkt objednal

## Vytvoření šablony / značky FAVI Partner Events Tracking

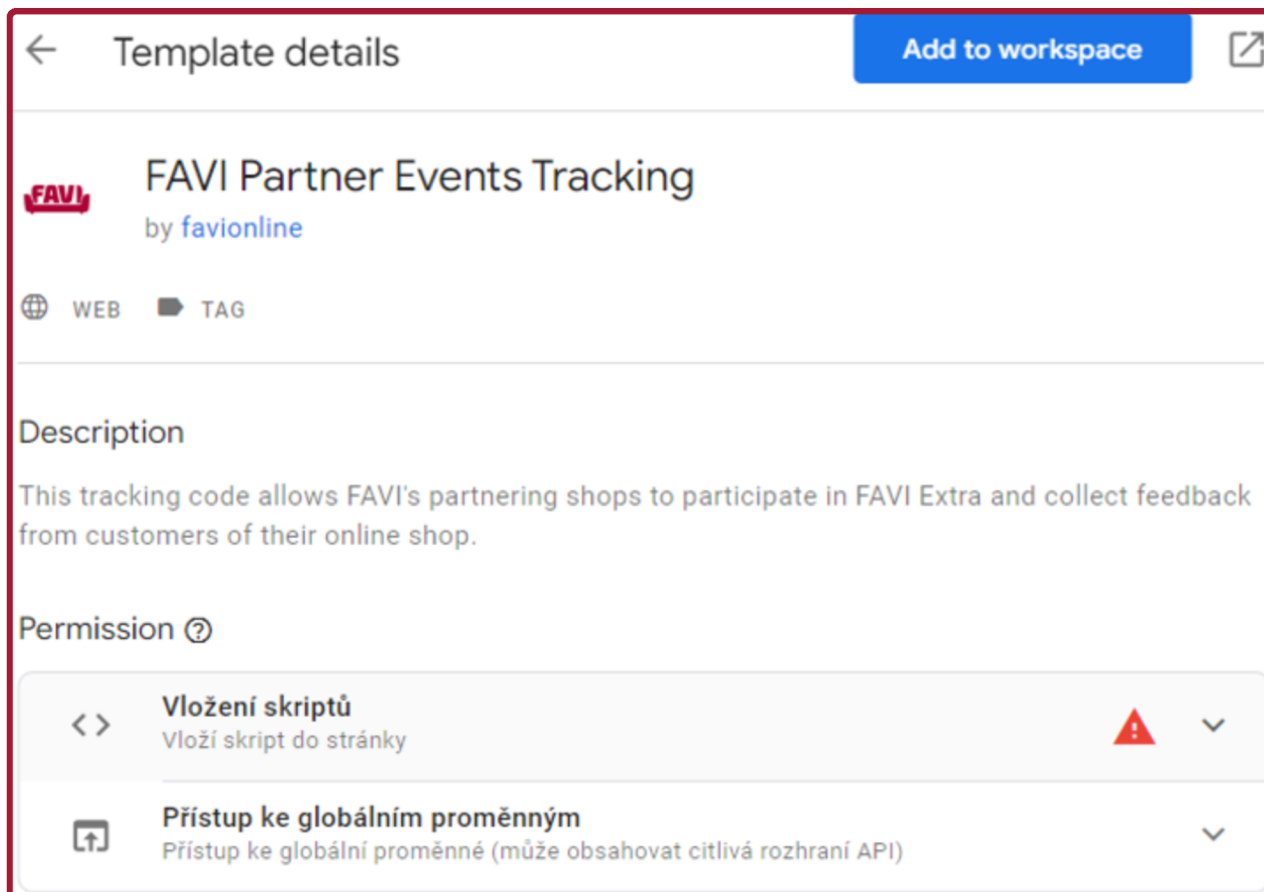
1.V GTM si vyberte preferovaný účet, kam budete implementovat **FAVI Partner Events Tracking** <https://tagmanager.google.com/?authuser=1#/home> a vyberte “New Tag”



2. Zde klikněte na “Tag Configuration” a vyberte “Discover more tag in the Community Template Gallery” a do vhledávacího pole napište slovo “favi” a vyberte “FAVI Partner Events Tracking”



3. Zde klikněte na modrý button “Add to workspace”. tím přidáte gallery template mezi vaše tagy



4. Do volného pole “**Tracking ID (required)**” zadejte identifikační klíč, který naleznete na odkazu, který jsme vám poslali pro přehled vašeho e-shopu

5. Do spodního volného pole napište “**{{js.FaviOrderEvents}}**” - jedná se o proměnnou, kterou jsme si vysvětlili výše (pokud se název rozhodnete přejmenovat jinak, není to problém, ale musí být shodná s tím, jak pojmenujete vlastní proměnnou pro JavaScript)

6. Do horního levého rohu zadejte název této značky, například “FAVI Partner Events Tracking”

FAVI Partner Events Tracking

Added in this workspace Abandon changes

Tag Configuration

Tag Type

**FAVI Partner Events Tracking**  
favionline GALLERY ✎

**Tag permissions** 2 permissions >

Read documentation (<https://github.com/favionline/partner-events-tracking-gtm-template/blob/master/doc/implementation-guide/readme.md>) for more detailed description and examples.

Tracking ID (required) ?

plz8gxuoavsartqfmj7ubtgbbxwm 🗑️

Event Type ?

createOrder ▼

createOrder Event Data (required) ?

{{js.FaviOrderEvents}} 🗑️

Vložit přidělené ID

Přidat vytvořenou proměnnou pro vlastní JavaScript

## 9. Tracking ID

- ID získáte ve vašem dashboardu v sekci “optimalizace kampaně” a záložce “Partner events tracking”, který vám byl vygenerovaný při registraci vašeho obchodu ve FAVI.
- Stránka s identifikačním číslem vypadá podobně takto

The screenshot displays a dashboard interface. At the top, a green banner indicates 'Obchod je aktivní'. Below it, a navigation menu includes 'Statistiky', 'Ceny za proklik', 'Optimalizace kampaně' (highlighted with a red arrow), 'Vypnutí kategorie', 'Fakturace', and 'Nastavení'. A sub-menu below 'Optimalizace kampaně' shows 'Partner events tracking' (highlighted with a red arrow) and other options like 'Optimalizace konverzí (GA4)', 'FAVI EXTRA', 'Kvalita produktových dat', and 'Slevové kódy'. A yellow warning banner states 'Partner events tracking je v testovacím režimu'. Under 'Ověřovací údaje', a table lists tracking information:

Tracking ID	Server-side token	Datum vytvoření
xxxxxxxxxxxxxxxxxxxxxxxxxx	xx	15.11.2024 11:19:52

Below this, a 'Statistika zaznamenaných objednávek' section shows a table with three columns: 'Počet zaznamenaných objednávek za posledních 24 hodin', 'Počet zaznamenaných objednávek za posledních 7 dní', and 'Celkový počet zaznamenaných objednávek'. All values are currently 0.

## Vytvoření šablony / značky FAVI Partner Events Tracking

1. Nyní přidejte “trigger” na kterém se má tato značka spouštět
  - Nastavte ho pouze na té stránce, kde je informace o úspěšně objednaného produktu (tzv. Thank You Page).
  - Viz první krok, kde jsme si ukazovali, kde se nám spouští Data Layer s celou objednávkou

2. Klikněte na “Triggering” a poté na plus vpravo nahoře (tím vytvoříte nový spouštěč)

FAVI Partner Events Tracking Save

Added in this workspace Abandon changes

### Tag Configuration

Tag Type

**FAVI Partner Events Tracking**  
favionline GALLERY

**Tag permissions** 2 permissions >

Read documentation (<https://github.com/favionline/partner-events-tracking-gtm-template/blob/master/doc/implementation-guide/readme.md>) for more detailed description and examples.

Tracking ID (required) ⓘ  
plz8gxuoavsartqfmj7ubtgbbxwm7mx1

Event Type ⓘ  
createOrder

createOrder Event Data (required) ⓘ  
{js.FAVIOrderEvents}

### Triggering

Choose a trigger to make this tag fire... Learn More

Choose a trigger

Name ↑	Type	Filter
All Pages	Page View	--
Click Elements - /basket/reg	All Elements	Page Path contains reg ⓘ
Consent Initialization - All Pages	Consent Initialisation	--



3. Zde vyberte tu možnost, kde se vám propisují potřebné atributy, (viz výše) o produktu a zákazníkovi

- Nejčastějším způsobem se atributy propisují na událost “purchase”
- V našem případě vybereme tedy “Custom event”
- Custom Event je zapotřebí správně nastavit na tu událost pod kterým se nám propisují potřebné hodnoty. V našem případě (a nejčastěji) se jedná o událost “purchase”

Event - purchase

Added in this workspace Abandon changes

Trigger Configuration

Trigger Type

Custom Event

Event name   Use regex matching

This trigger fires on

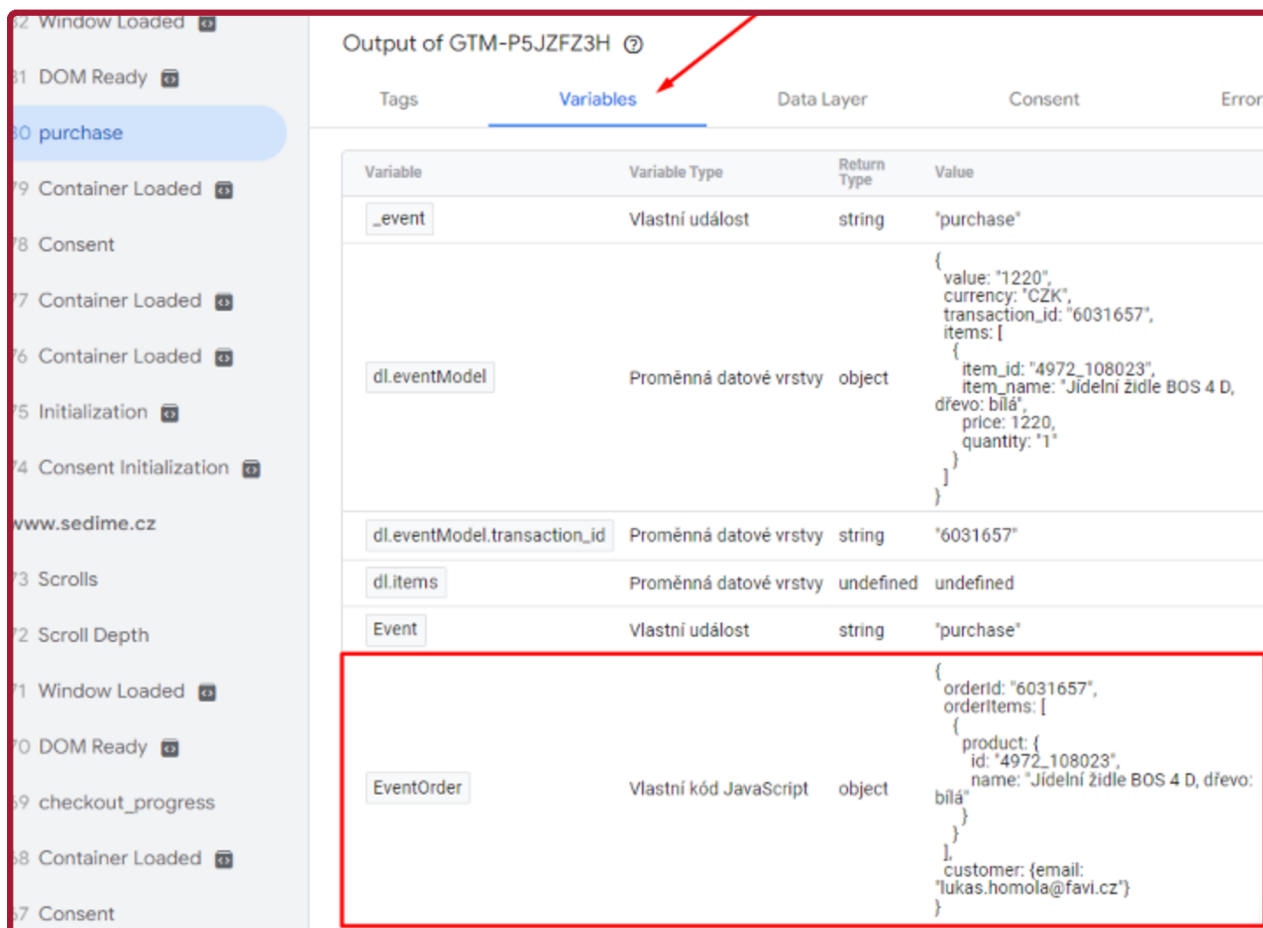
All Custom Events  Some Custom Events

4. A nyní celý tag uložte kliknutím na modré tlačítko “Uložit”

## Testování změn a ověření správnosti našich proměnných

1. Nyní přes “**preview**” udělejte novou objednávku a proveďte celý proces nákupu znovu. Musíte dojít až na stránku, kde je úspěšně produkt zakoupen (Thank You Page)

2. Pokud máte vše nastavené správně, tak ve “**variables**” pod názvem vlastní proměnné “**js.FAVIOrderEvents**” uvidíte dynamické hodnoty produktu a kupujícího, které jste vyplnili ve formuláři vaší testovací objednávky



Output of GTM-P5JZFZ3H

Tags Variables Data Layer Consent Errors

Variable	Variable Type	Return Type	Value
_event	Vlastní událost	string	"purchase"
dl.eventModel	Proměnná datové vrstvy	object	{ value: "1220", currency: "CZK", transaction_id: "6031657", items: [ { item_id: "4972_108023", item_name: "Jídelní židle BOS 4 D, dřevo: bílá", price: 1220, quantity: "1" } ] }
dl.eventModel.transaction_id	Proměnná datové vrstvy	string	"6031657"
dl.items	Proměnná datové vrstvy	undefined	undefined
Event	Vlastní událost	string	"purchase"
EventOrder	Vlastní kód JavaScript	object	{ orderid: "6031657", orderitems: [ { product: { id: "4972_108023", name: "Jídelní židle BOS 4 D, dřevo: bílá" } } ], customer: {email: "lukas.homola@favi.cz"} }

## Závěr, užitečné odkazy a nejčastější chyby

### Závěr:

Doufáme, že tento návod alespoň trochu ulehčil práci s implementací **FAVI Extra**.

V každém případě vám rádi pomůžeme a v takovém případě nás neváhejte kontaktovat na [lukas.homola@favi.cz](mailto:lukas.homola@favi.cz)

Zároveň prosíme o sdílení **GTM účtu s právy EDIT** na e-mail [partner-events-tracking-support@favionline.com](mailto:partner-events-tracking-support@favionline.com)

### Nemáte nikde potřebné údaje o objednavce a zákazníkovi?

- Abyste mohli FAVI Extra implementovat úspěšně je zapotřebí, abyste měli údaje o objednavce a zákazníkovi. K tomu doporučuji implementovat DL Ecommerce strukturu (jednoduchou) na stránce, kde je shrnutí objednávky po úspěšném objednání (Thank You Page)
- Vše je vysvětleno v této [Google příručce](#)

### Užitečné odkazy:

- Zde je [úplná dokumentace](#) pro implementaci, kde je i manuál napojení FAVI Extra nejen přes GTM
- Dokumentace v Githubu pro implementaci FAVI Extra přes GTM je [zde](#)
- [Google nápověda](#) pro spouštěče (triggery)
- [Google nápověda](#) pro proměnné (variables)

- [Google nápověda](#) pro DataLayer
- [Google nápověda](#) pro testování GTM
- [Gogole nápověda](#) pro správnou implementaci GTM kódu

## Chyby v GTM:

- Pokud se vám vyskytla chyba v GTM při náhledu s textem:
  - “Neznámá proměnná "xxx" nalezena v jiné proměnné. Upravte proměnnou a odstraňte odkaz na neznámou proměnnou.”
  - Možné příčiny:
    - Tam, kde jste ji použili pomocí znaky “{{ }}” je špatně napsaná, než jak je reálně uložené pod nabídkou “proměnné”
    - Nebo není vůbec vytvořena, ale snažíte se na ní cílit
- Pokud se Vám vyskytla chyby v GTM při náhledu s textem:
  - “Chyba na řádce 1, znak 11: Chyba analýzy. ')' očekáváno”
  - Možné příčiny:
    - Zaměřte se na daný řádek a zkuste vyřešit závorky, jelikož zřejmě někde chybí
    - **POZOR: Vlastní proměnná musí být vždy tvořena pomocí “function” a zároveň očekává “return”**

