



Grav



tutorialspoint

S I M P L Y E A S Y L E A R N I N G

www.tutorialspoint.com



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

About the Tutorial

Grav is a flat-file based content management system which doesn't use database to store the content instead it uses text file (.txt) or markdown (.md) file to store the content. The flat-file part specifically refers to the readable text and it handles the content in an easy way which can be simple for a developer.

Audience

This tutorial has been prepared for anyone who has a basic knowledge of Markdown and has an urge to develop websites. After completing this tutorial, you will find yourself at a moderate level of expertise in developing websites using Grav.

Prerequisites

Before you start proceeding with this tutorial, we assume that you are already aware about the basics of Markdown. If you are not well aware of these concepts, then we will suggest you to go through our short tutorials on Markdown.

Copyright & Disclaimer

© Copyright 2017 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com

Table of Contents

About the Tutorial.....	i
Audience	i
Prerequisites	i
Copyright & Disclaimer.....	i
Table of Contents	ii
1. GRAV – OVERVIEW	1
2. GRAV – INSTALLATION.....	3
Software Requirements for Grav.....	3
Download Grav	3
Setup Wizard.....	4
3. GRAV – PAGES.....	7
Page Types	8
Folders	10
Ordering.....	11
Page File.....	12
Summary Size and Separator.....	12
Finding other Pages.....	13
4. GRAV – MARKDOWN SYNTAX.....	14
Headings	15
Comments.....	15
Horizontal rules.....	16
Body Copy	16
Emphasis	17
Notices	19

Lists.....	22
Code.....	24
Tables.....	25
Links.....	27
Add a Title.....	28
Images.....	28
5. GRAV – PAGE LINKING.....	30
Slug Relative.....	31
Directory Relative	32
Absolute Links.....	33
Remote	35
6. GRAV – IMAGE LINKING.....	37
Slug Relative.....	38
Directory Relative	39
Absolute.....	40
Remote	41
Media Actions on Images	42
7. GRAV – MEDIA.....	43
Supported Media Files	43
Display Modes.....	43
Locating Thumbnails	43
Lightboxes and Links	44
Actions	44
Grav – Media URL.....	45
Grav - Media html	45

Grav - Media Link	46
Grav - Media Lightbox	47
Grav - Media Thumbnails	48
Grav - Media Resize.....	50
Grav - Media ForceResize	51
Grav - Media CropResize	51
Grav - Media Crop Images	53
Grav - Media CropZoom	54
Grav - Media Quality	55
Grav - Media Negation	56
Grav - Media Brightness	57
Grav - Media Contrast	58
Grav - Media Grayscale	59
Grav - Media Embossing Images.....	60
Grav - Media Smooth	61
Grav - Media Sharpness	62
Grav - Media Edge	63
Grav - Media Colorizing	64
Grav - Media Sepia	65
Grav - Media Re-size	66
Grav - Media High Density.....	68
Grav - Media Queries	69
Metafiles.....	70
8. GRAV – MODULAR PAGES	71
Folder Structure	71
How to Create Modular Page	72

9.	GRAV – MULTI LANGUAGE	75
	Multi – Languages Basics	75
10.	GRAV – THEMES BASICS	83
	Content Pages & Twig Templates	83
	Theme Organization	83
	Theme Example	87
11.	GRAV – THEME TUTORIAL	88
12.	GRAV – TWIG FILTERS & FUNCTIONS.....	100
	Twig Filters.....	100
	Twig Functions	106
13.	GRAV – THEME VARIABLES	108
	Core Objects.....	108
	pages object	116
	uri object.....	116
	Adding Custom Variables	119
14.	GRAV – ASSET MANAGER	120
	Configuration	120
	Assets in Themes.....	120
	Adding Assets.....	122
	Options	123
	Rendering Assets.....	124
	Named Assets	124
	Grouped Assets	125
	Static Assets	125

15. GRAV – THEME CUSTOMIZATION	126
Custom CSS	126
Custom SCSS/LESS	126
Theme Inheritance	127
16. GRAV – PLUGIN BASICS	131
17. GRAV – PLUGIN TUTORIALS.....	132
Setup Plugin	132
Plugin Configuration.....	132
Plugin Structure	133
Displaying Random Page	134
18. GRAV – EVENT HOOKS.....	137
Twig Event Hooks	138
Collection Event Hooks.....	138
Page Event Hooks.....	139
19. GRAV – ADMIN INTRODUCTION	141
Features	141
Installation	141
Manual Installation	142
Creating User	142
Usage	144
20. GRAV – ADMIN DASHBOARD.....	145
Maintenance and Statistics	147

21. GRAV - CONFIGURATION SYSTEM.....	149
Content	149
Languages	151
HTTP Headers	152
Markdown	153
Caching	154
Twig Templating	155
Assets	156
Error Handler	157
Debugger	157
Media	158
Session	159
Advanced	159
22. GRAV – CONFIGURATION SITE.....	160
23. GRAV – ADMINISTRATION PANEL PAGES.....	163
Adding New Pages	163
Pages List	166
24. GRAV – PAGE EDITOR OPTIONS.....	168
Publishing	168
Taxonomies	169
25. GRAV – PAGE EDITOR ADVANCED	170
Settings	171
Ordering	173
Overrides	173

26. GRAV – BLUEPRINTS.....	176
Resource Identity	176
Forms	179
Form Fields available in the admin	183
27. GRAV – PERFORMANCE & CACHING.....	185
Performance	185
Caching	185
Caching Types	186
Memcache Options	187
Redis Options	187
Caching and Events	188
28. GRAV – DEBUGGING & LOGGING	189
PHP Debug Bar	189
Error Display	191
Logging	191
29. GRAV – CLI.....	193
New Project Creation	193
Installing Grav Dependencies	194
Clearing Grav Cache	194
Create Backup	194
Updating Composer	195
30. GRAV – GPM.....	196
How Does it work?	197

31. GRAV – DEVELOPMENT	202
Theme / Plugin Release Process	203
32. GRAV – LIFECYCLE	205
33. GRAV – YAML SYNTAX	208
Features	208
Basic Rules of YAML	208
Basic Data Types of YAML	208
34. GRAV – FORMS.....	211
Creating a Simple Form	211
Fields Parameter	214
Note on Captcha	215
Form Actions	216
35. GRAV – WEB HOSTING	220
Rochen Web Hosting	220
WireNine.....	221
Crucial Web Hosting.....	222
Arvixe.....	223
SiteGround.....	224
Dreamhost	225
36. GRAV – SERVER ERROR.....	227
37. GRAV – PERMISSION	228

1. GRAV – OVERVIEW

Grav is a flat-file based content management system. This system does not use database to store the content, instead it uses a text (.txt) file or a markdown (.md) file to store the content. The flat-file part refers to the readable text and it handles the content in an easy way which can be simple for a developer.

Grav was developed by the **Rocket Theme** team. It runs on PHP and is an open-source CMS like Joomla, Wordpress or Drupal.

Why Grav?

Grav is now the fastest flat-file content management system. It is easier to publish and manage content with Grav. It allows a developer to handle content very easily by storing content in files (such as text or markdown files) rather than in database.

Grav helps you build a simple, manageable and a quick site. It costs less than the database driven CMS and is useful with I/O for file handling, when you don't have enough resources to handle database.

Features of Grav

- It is a fast, easy and powerful flat-file web platform.
- It is used to build websites with no extra tools or html knowledge.
- It uses text file or markdown file to store the content.
- It doesn't use database, so this mitigates the instances of bottlenecking.
- It uses PHP based template provided by Twig which is parsed directly into PHP that makes it fast.

Advantages

- It is not based on database CMS, so it's very easy to install and will be ready to use when you upload the files to the server.
- It uses Markdown text files to make things easy. Using this, the content is dynamically converted to HTML and displayed in the browser.
- In case of security, Grav doesn't have admin area and database. So there is no chance of hacking into account or in the database to access the important data.
- You can easily backup all the files to keep the backup copy of your website, since there is no database to backup.
- It is a piece of software which doesn't require more time to learn.

Disadvantages

- You might come across instances where unauthorized users may access your content from the files directly as there is no database for the files.
- It is difficult to build complex websites using Grav CMS.

2. GRAV – INSTALLATION

In this chapter, we will understand the installation of Grav. We will discuss the software requirements for Grav and also how to download it.

Software Requirements for Grav

Let us now understand the software requirements for Grav.

Web Server

- WAMP (Windows)
- LAMP (Linux)
- XAMP (Multi-platform)
- MAMP (Macintosh)
- Nginx
- Microsoft IIS

Operating System: Cross-platform

Browser Support: IE (Internet Explorer 8+), Firefox, Google Chrome, Safari, Opera

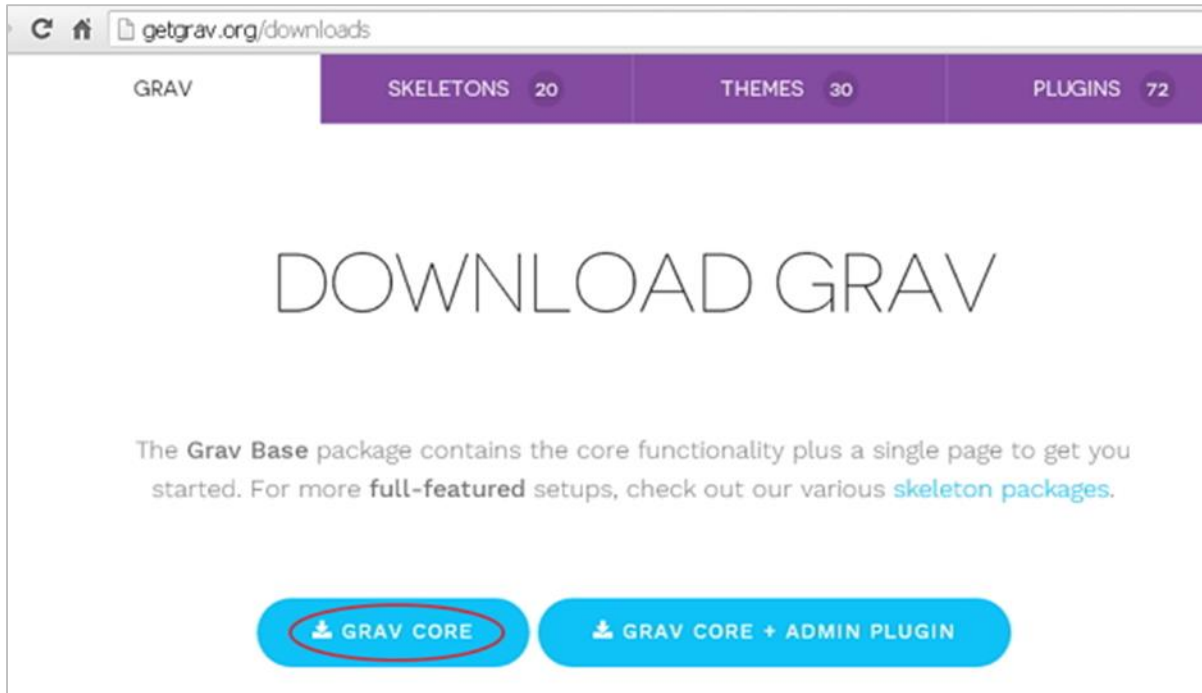
PHP Compatibility: PHP 5.4 or higher

Text Editors:

- Sublime Text (Mac / Windows/ Linux)
- Atom (Mac / Windows)
- Notepad ++ (Windows)
- Bluefish (Mac / Windows/ Linux)

Download Grav

Click on this link <http://getGrav.org/downloads> and follow the steps as shown in the screenshot given below to download Grav.

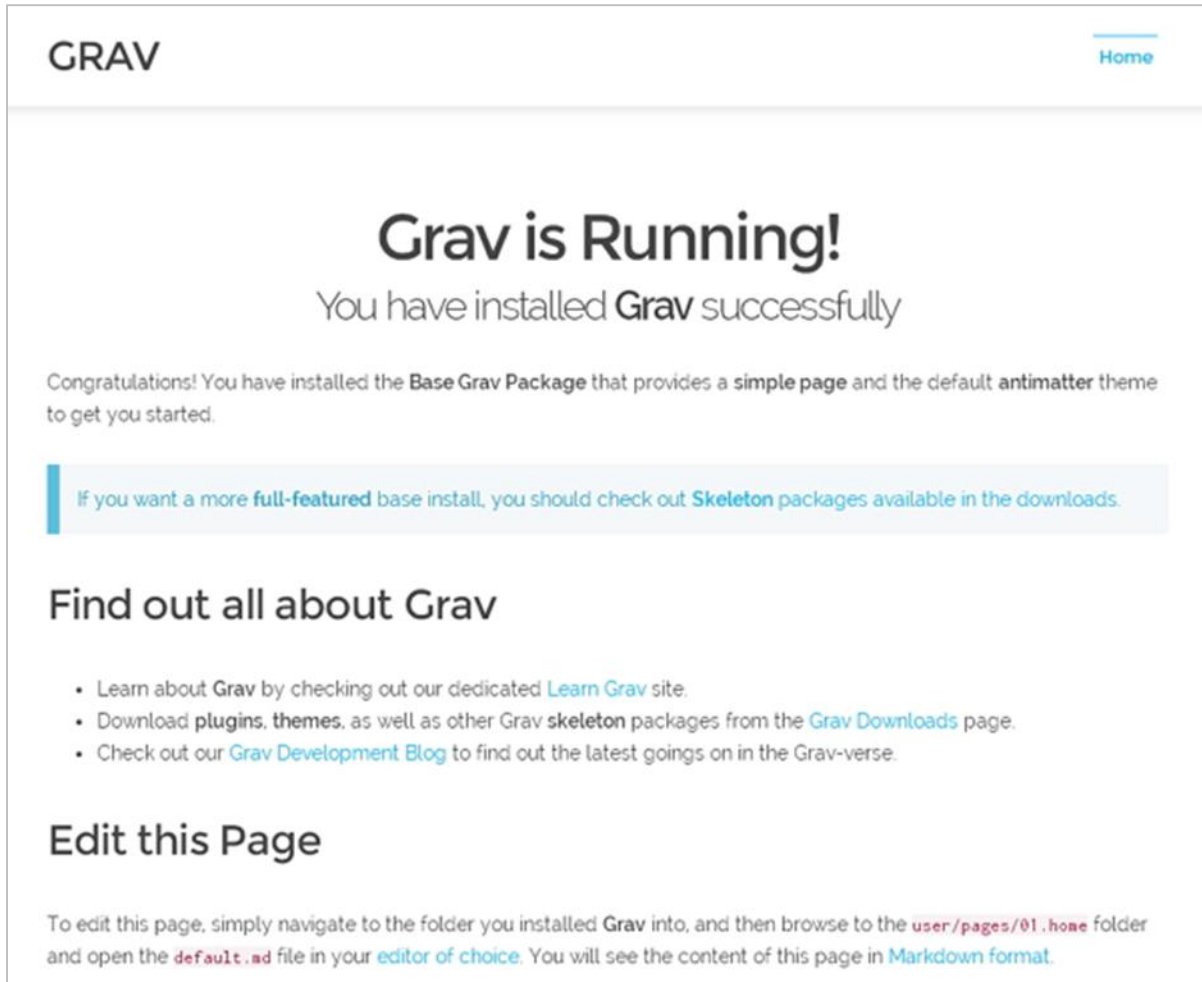


Unzip the downloaded Grav file into your web server.

Setup Wizard

Installation of Grav is a very simple process. Follow the steps given below for Grav setup.

- Download the zip file and extract it to your web server or local host. Rename the folder from its current name that you want to use to refer to your site.
- Open your browser and navigate to **localhost/<your_folder_name>**, you will be redirected to a screen which shows **you have installed Grav successfully** as in the following screenshot.



The screenshot shows the Grav installation success page. At the top left is the word "GRAV" and at the top right is a "Home" link. The main heading is "Grav is Running!" followed by "You have installed Grav successfully". Below this is a congratulatory message: "Congratulations! You have installed the Base Grav Package that provides a simple page and the default antimatter theme to get you started." A light blue callout box contains the text: "If you want a more full-featured base install, you should check out Skeleton packages available in the downloads." The next section is "Find out all about Grav" with a bulleted list: "Learn about Grav by checking out our dedicated Learn Grav site.", "Download plugins, themes, as well as other Grav skeleton packages from the Grav Downloads page.", and "Check out our Grav Development Blog to find out the latest goings on in the Grav-verse." The final section is "Edit this Page" with instructions: "To edit this page, simply navigate to the folder you installed Grav into, and then browse to the user/pages/01.home folder and open the default.md file in your editor of choice. You will see the content of this page in Markdown format."

- Grav comes with a sample page that helps you get started. In the above screenshot, you can see the home link which has displayed a sample page.

Grav Content

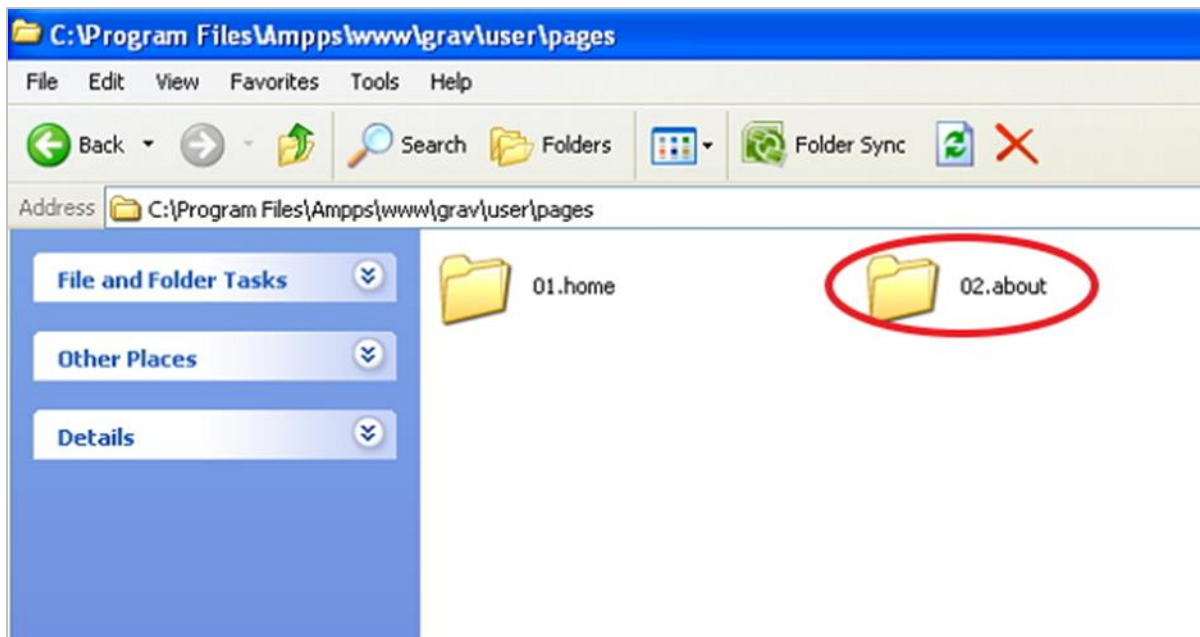
3. GRAV – PAGES

In this chapter, let us study about **Grav Pages**. Pages can be defined as building blocks of the site. Pages combine contents and navigations; this makes work easier even for the inexperienced users.

To begin with, let us know how to create a simple page. All user contents will be stored under **user/pages/** folder. There will be only one folder called **01.home**. The numeric portion of the folder is optional; it expresses the order of your pages (for example, 01 will come before 02) and explicitly informs Grav that this page should be visible in menu.

Let us now see how to create a new page.

Step 1: Create a folder under **/user/pages/**; for example, **02.about** as shown in the following screenshot.



Step 2: Create a file called **default.md** inside the newly created **02.about** folder with the following content.

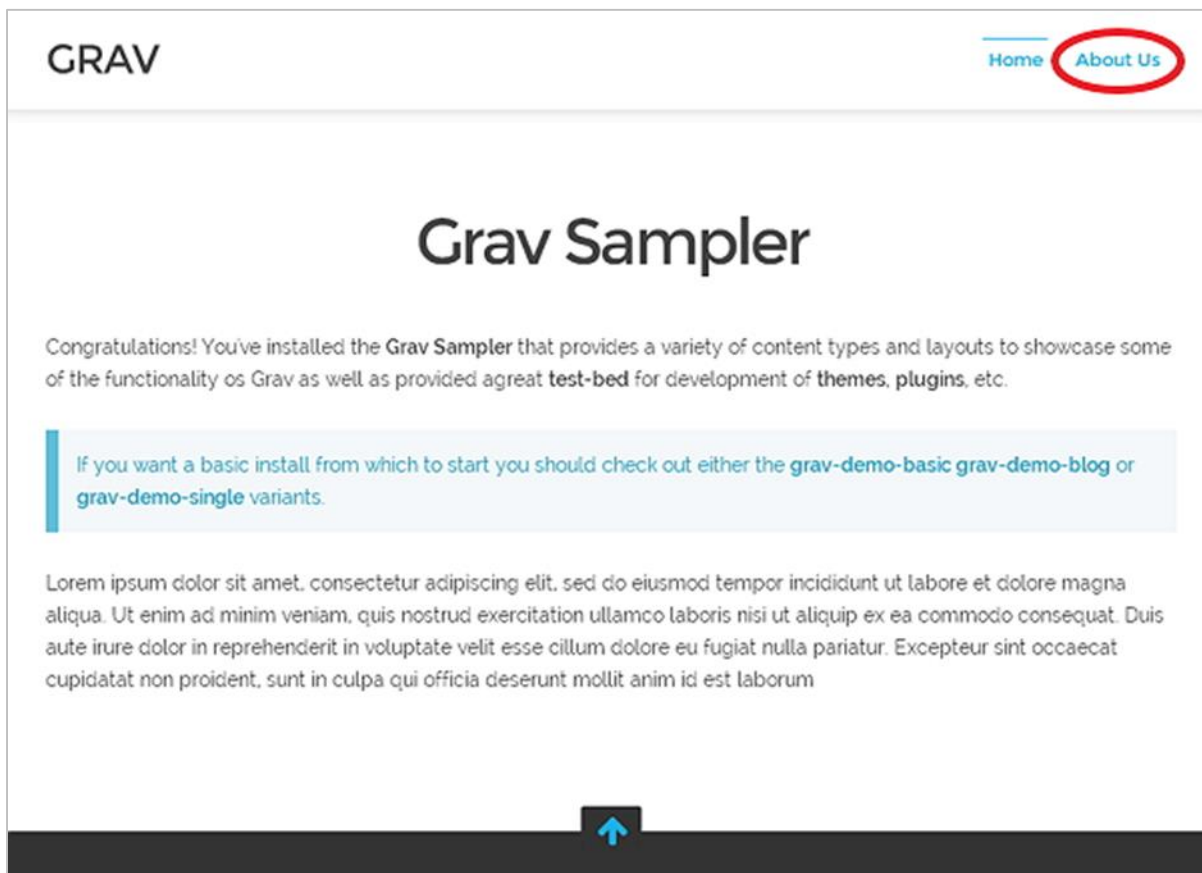
```
---  
title: About Us  
---  
# About Us Page!
```

```
This is the body of about us page.
```

The above code uses some **Markdown** syntax explained briefly below. You can study in detail about **Markdown** in [Markdown](#) chapter.

- The content between the `---` indicators are the **Page Headers**.
- **#** or **hashes** syntax in **Markdown** indicates a title which will be converted to `<h1>` header in HTML.
- ****** markers indicates bold text or `` in HTML.

Step 3: Reload your browser and you can see new page in menu as shown in the following screenshot.



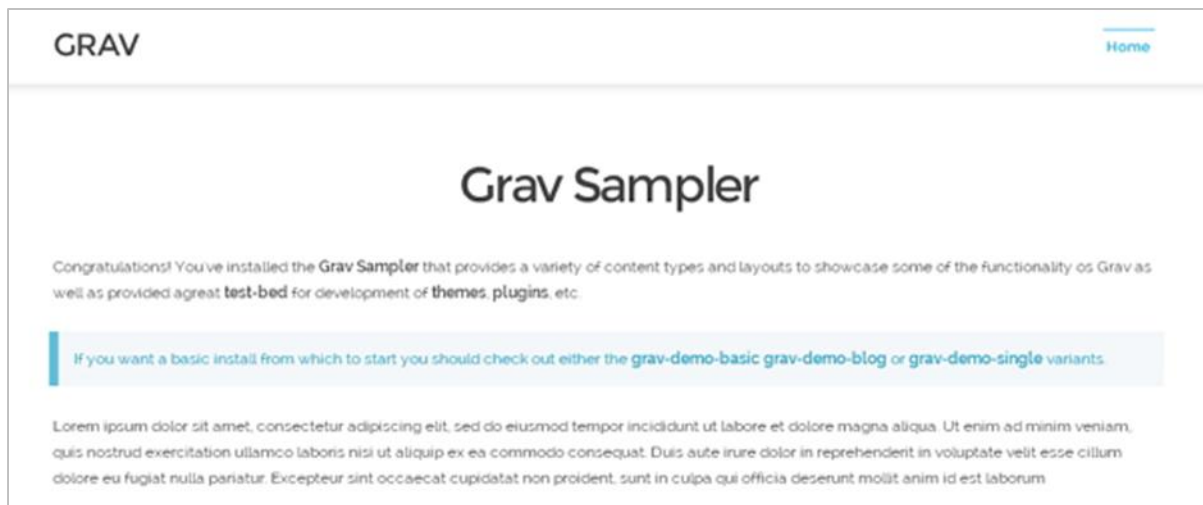
Page Types

Grav Pages supports 3 types of pages:

- Standard Page.
- Listing Page.
- Modular Page.

Standard Page

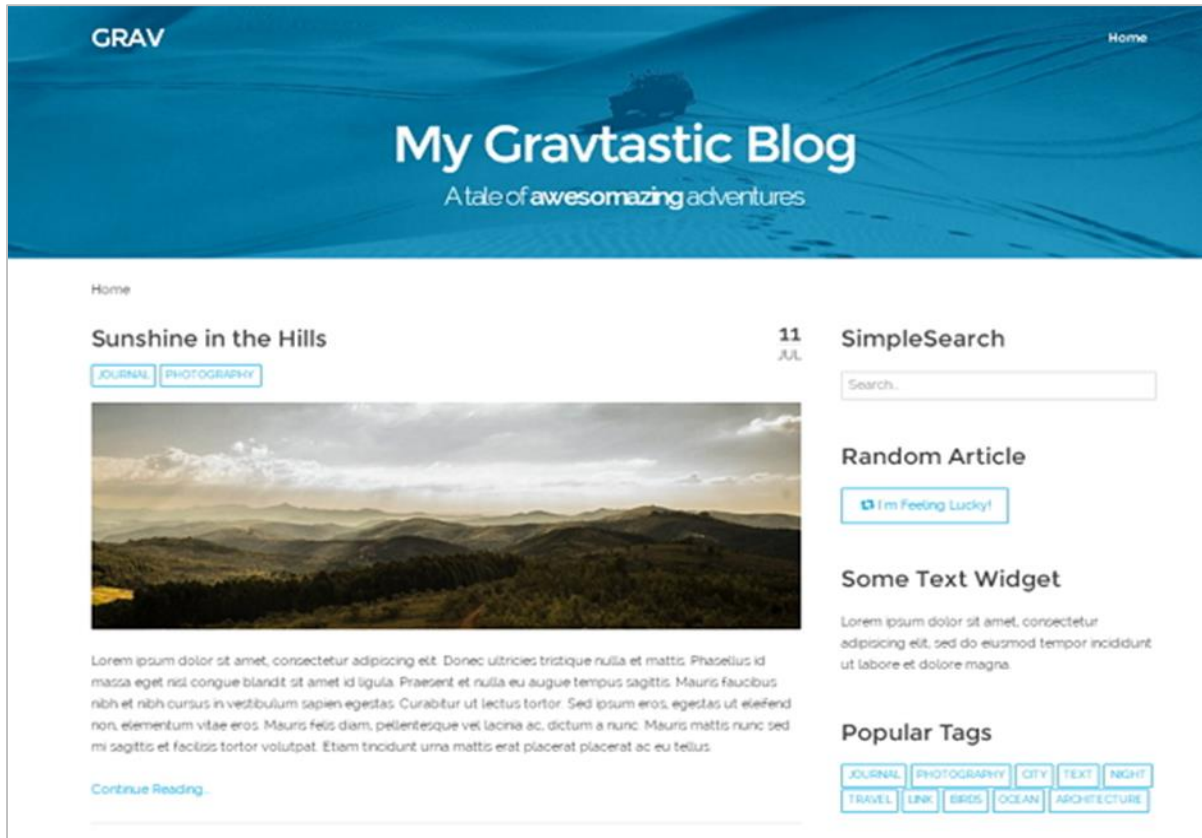
Standard Pages are most basic type of pages such as blog post, contact form, error page etc. By default, a page is considered as a Standard Page. You are welcomed by a Standard Page as soon as you download and install the Base Grav package. You will see the following page when you install Base Grav package.



Listing Page

Listing Page is an extension of a standard page which has a reference to a collection of pages. The easiest way to set up the listing page is to create child pages below the listing page. A blog listing page is a fine example for this.

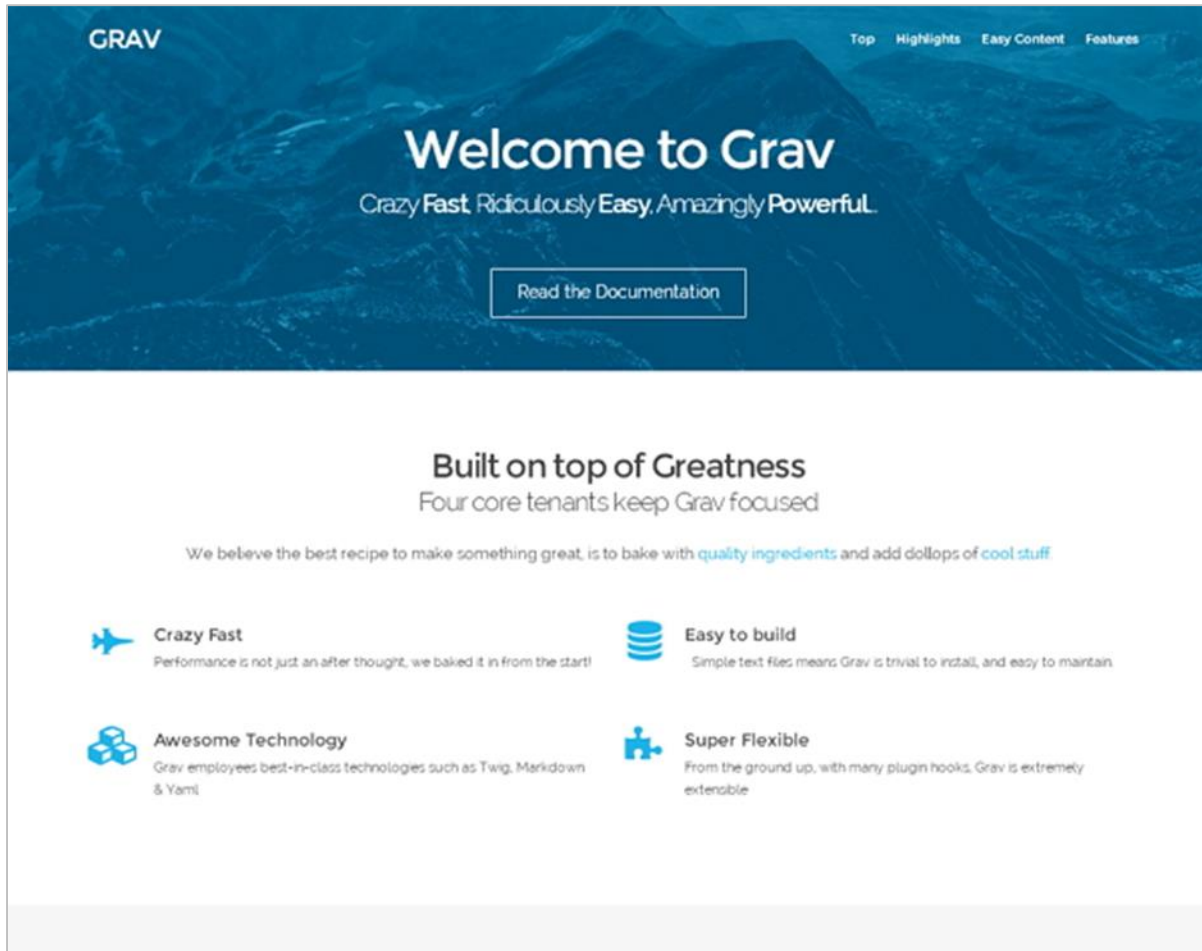
A sample Blog Skeleton with Listing Page can be found in the [Grav Downloads](#). A sample one is shown in the following screenshot.



Modular Page

Modular Page is a form of listing page which builds a single page from its child pages. This allows us to build very complex one-page layouts from smaller modular content pages. This can be achieved by building the modular page from multiple modular folders found in the page's primary folder.

A sample one-page skeleton using a Modular Page can be found in the [Grav Downloads](#). A sample one is shown in the following screenshot.



The image shows the Grav website landing page. At the top left is the 'GRAV' logo. On the top right are navigation links: 'Top', 'Highlights', 'Easy Content', and 'Features'. The main heading is 'Welcome to Grav' with the tagline 'Crazy Fast. Ridiculously Easy. Amazingly Powerful.' Below this is a button that says 'Read the Documentation'. The section below is titled 'Built on top of Greatness' with the subtitle 'Four core tenants keep Grav focused'. A paragraph states: 'We believe the best recipe to make something great, is to bake with quality ingredients and add dollops of cool stuff'. There are four feature cards: 'Crazy Fast' (with a jet icon) stating 'Performance is not just an after thought, we baked it in from the start!'; 'Easy to build' (with a stack of disks icon) stating 'Simple text files means Grav is trivial to install, and easy to maintain'; 'Awesome Technology' (with a cube icon) stating 'Grav employs best-in-class technologies such as Twig, Markdown & Yaml'; and 'Super Flexible' (with a puzzle piece icon) stating 'From the ground up, with many plugin hooks, Grav is extremely extensible'.

Folders

The **/user/pages** folder will contain contents for their respective pages. The folders inside the **/user/pages** folder are automatically treated as menus by Grav and used for the purpose of ordering. For example, the **01.home** folder will be treated as home. Ordering is also to be maintained, i.e, 01.home will come before 02.about.

You should provide an entry-point so that it specifies the browser where to go when you point browser to root of your site. For example, if you enter `http://mysite.com` in your browser, Grav expects an alias **home/** by default, but you can override the home location by changing the **home.alias** option in the Grav configuration file.

Underscore (`_`) before the folder name is identified as **Modular folders**, which is a special folder type that is intended to be used only along modular content. For example, for the folder such as **pages/02.about**, slug would default to **about**, and the URL will be `http://mysite.com/about`.

If the folder name is not prefixed with numbers, that page is considered to be invisible and will not be displayed in navigation. For example, the if **user/pages/** has **/contact** folder, will not be displayed in navigation. This can be overridden in the page itself inside the header section by setting visible to true as shown below to make it visible in navigation.

```
---
title: contact
visible: true
---
```

By default, a page is visible in the navigation if the surrounding folders have numerical prefixes. The valid values for setting visibility are **true** or **false**.

Ordering

There are many ways to control ordering of the folder, one of the important way is to set **content.order.by** of the page configuration settings. The options are listed below.

- **default:** File system can be used for ordering, i.e., 01.home before 02.about.
- **title:** Title can be used for ordering which is defined in each page.
- **date:** Ordering can be based on date which is defined in each page.
- **folder:** Folder name consisting of any numerical prefix, e.g. 01., will be removed.
- **basename:** Ordering is based on the alphabetic folder without numeric order.
- **modified:** Modified timestamp of the page can also be used.
- **header.x:** Any of the page header field can be used for ordering.
- **manual:** Using order_manual variable ordering can be made.
- **random:** Randomizing your order can also be done.

Manual order is specifically defined by providing a list of options to the **content.order.custom** configuration setting. You can set the **pages.order.dir** and the **pages.order.by** options to override the default behavior in the Grav system configuration file.

Page File

The page inside the page folder should be created as **.md** file, i.e., Markdown formatted file; it is markdown with YAML front matter. The **default** will be the standard name for main template and you can give it any name. An example for a simple page is shown below:

```

---
title: Title of the page
taxonomy:
  category: blog page
---
# Title of the page

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque porttitor eu felis sed ornare. Sed a mauris venenatis, pulvinar velit vel, dictum enim. Phasellus ac rutrum velit. **Nunc lorem** purus, hendrerit sit amet augue aliquet, iaculis ultricies nisl. Suspendisse tincidunt euismod risus. Nunc a accumsan purus.
```

Contents between --- markers is known as the YAML front matter and this YAML front matter consists of basic YAML settings. In the above example, we are setting title and taxonomy to the blog page. The section after the pair of --- markers is the actual content that we see on our site.

Summary Size and Separator

The default size of the summary can be set in **site.yaml** used via **page.summary()**. This is useful for blogs where just the summary information is needed and not the full page content. You can use the **manual summary separator** also known as **summary delimiter: ===** and ensure you put this in your content with blank lines above and below it, as shown below.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

===

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
```

```
pariat. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."
```

The text above the separator will be used when referenced by **page.summary()** and the full content when referenced by **page.content()**.

Finding other Pages

Grav has feature called **find()** method to find another page and perform actions on that page.

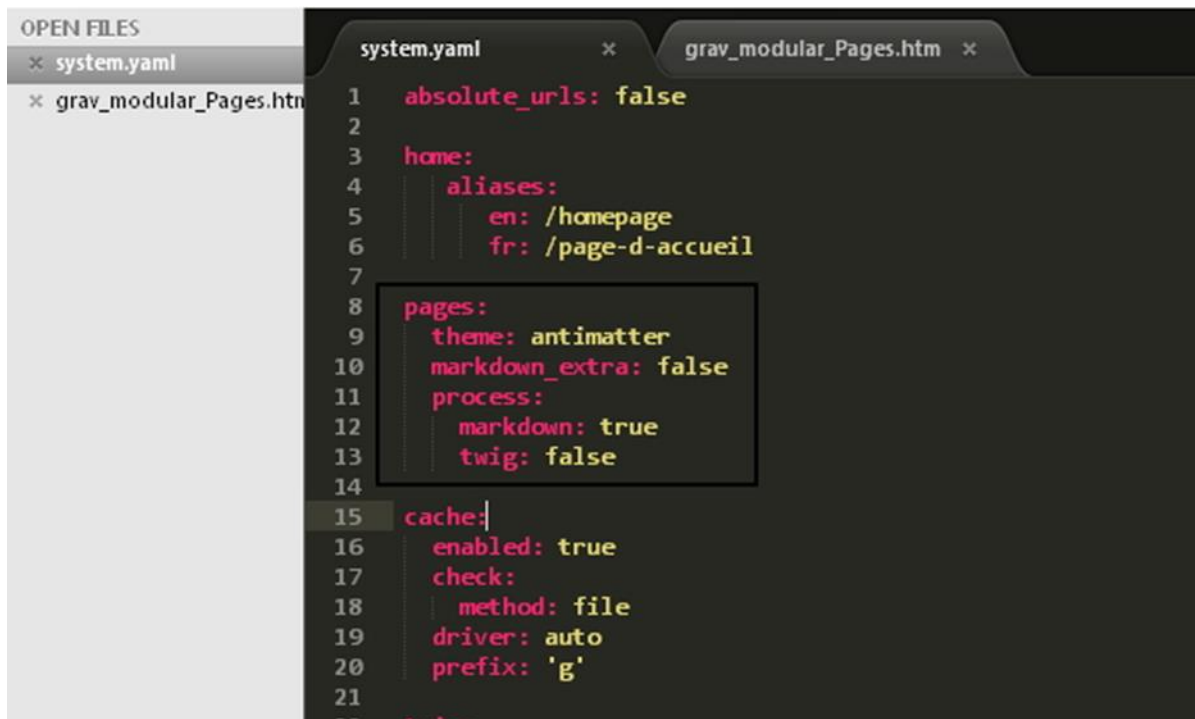
For example, if you want to list all the company location on a particular page, use the following markdown rule:

```
# Locations
<ul>
{% for loc in page.find('/locations').children if loc != page %}
<li><a href="{{loc.url}}">{{ loc.title }}</a></li>
{% endfor %}
</ul>
```


4. GRAV – MARKDOWN SYNTAX

Markdown syntax is defined as writing plain text in an easy to read and easy to write format, which is later converted into HTML code. Symbols like (*) or (`) are used in markdown syntax. These symbols are used to bold, creating headers and organize your content.

To use Markdown syntax, you must create a `.md` file in your `user/pages/02.mypage` folder. Enable Markdown Syntax in your `\user\config\system.yaml` configuration file.



```
1 absolute_urls: false
2
3 home:
4   aliases:
5     en: /homepage
6     fr: /page-d-accueil
7
8 pages:
9   theme: antimatter
10  markdown_extra: false
11  process:
12    markdown: true
13    twig: false
14
15 cache:
16   enabled: true
17   check:
18     method: file
19   driver: auto
20   prefix: 'g'
```

There are many benefits of using Markdown syntax, some of them are as follows.

- It is easy to learn and has minimum characters.
- When you use markdown there are very few chances of having errors.
- Valid XHTML output.
- Your content and visual display is kept separate so that it does not affect the look of your website.
- You can use any text editor or markdown application.

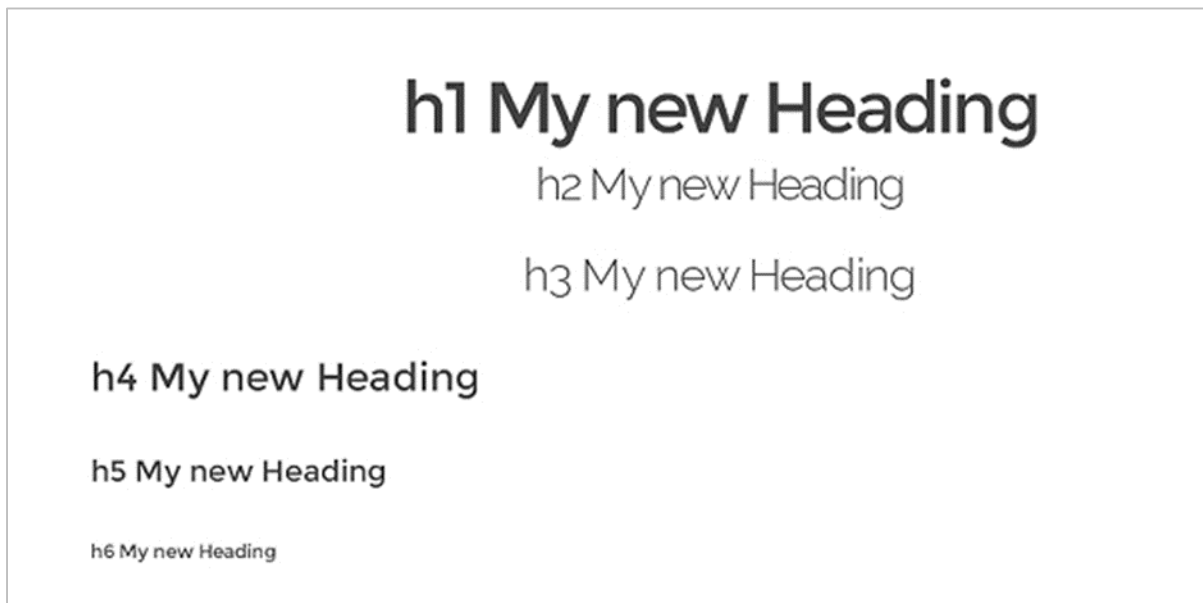
In the following sections, we will discuss the main elements of HTML that are used in markdown.

Headings

Each heading tag is created with # for each heading, i.e., from h1 to h6 the number of # increases as shown.

```
#my new Heading
##my new Heading
###my new Heading
####my new Heading
#####my new Heading
#####my new Heading
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Comments

You can write comments in the following format.

```
<!--
```

```
This is my new comment  
-->
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:

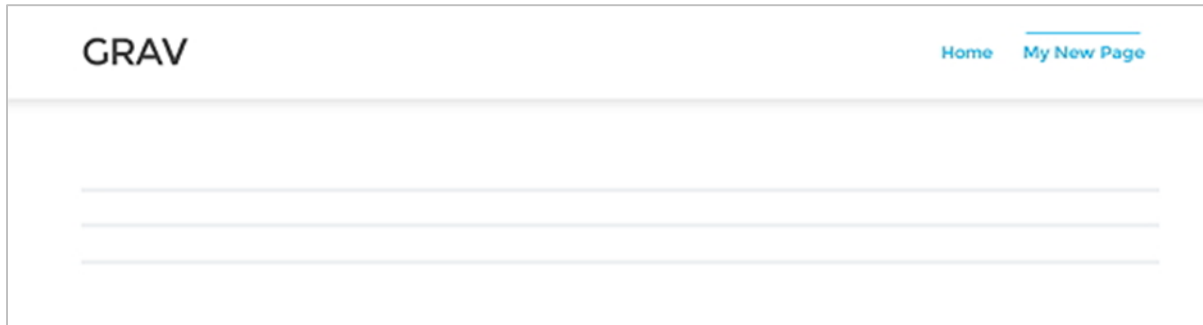
Appears Blank

Horizontal rules

Horizontal rules are used to create a thematic break in between paragraphs. You can create breaks between paragraphs using any of the following methods.

- `___`: Three underscores
- `---`: Three dashes
- `***`: Three asterisks

Open the **md** file in a browser as **localhost/Grav/mypage**; you will receive the following result:



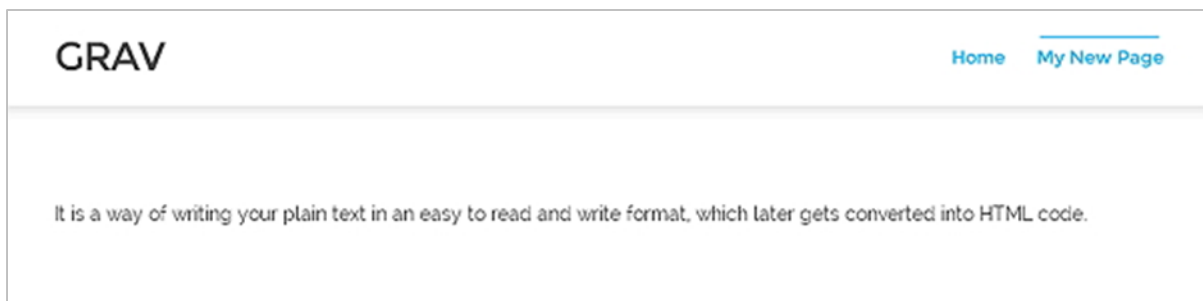
Body Copy

Body copy can be defined as writing text in normal format in markdown syntax, no **(p)** tag is used

Example

```
It is a way of writing your plain text in an easy to read and write format, which later gets converted into HTML code.
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Emphasis

Emphasis are the writing formats in markdown syntax that are used to bold, italicize or strikethrough a portion of text. Let us discuss them below:

Bold

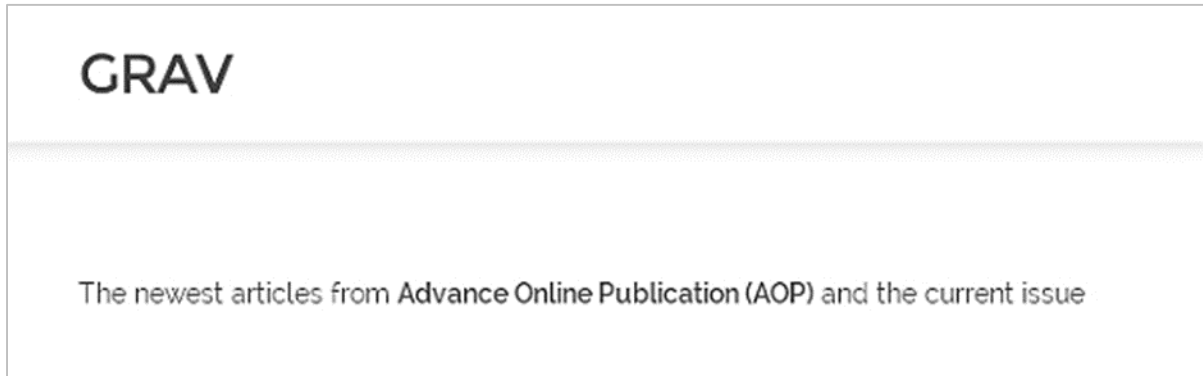
A portion of text can be made bold using two **(**)** signs at either sides.

Example

The newest articles from **Advance Online Publication (AOP)** and the current issue.

In this example, we have to show '**Advance Online Publication (AOP)**' word as bold.

Open the `.md` file in a browser as `localhost/Grav/mypage`, you will receive the following result:



Italics

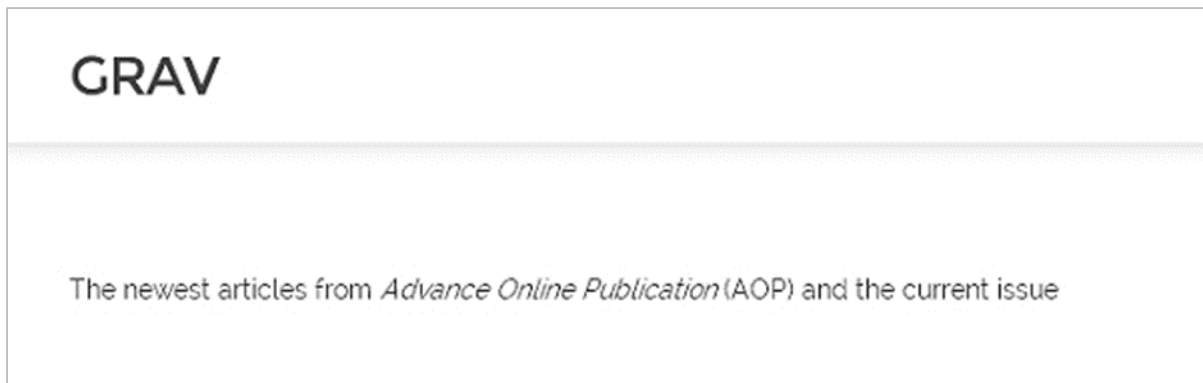
Use “`_`” (underscores) sign at either sides of the word to italicize the text.

Example

The newest articles from *Advance Online Publication* (AOP) and the current issues.

In this example, we have to italicize “**Advance Online Publication**” (AOP) word.

Open the `.md` file in a browser as `localhost/Grav/mypage`. This will give you the following result:



Strikethrough

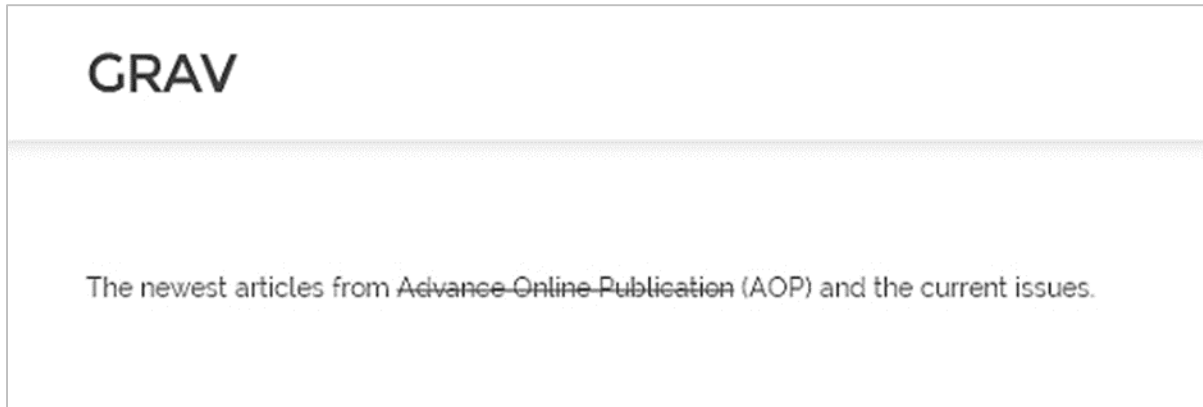
Use two “`~`” (tildes) on either sides of the word to strikethrough the word.

Example

```
The newest articles from ~~Advance Online Publication~~ (AOP) and the current issues.
```

In this example, we have to strike "Advance Online Publication" (AOP) word.

Open the `.md` file in a browser as **localhost/Grav/mypage..** This will give you the following result:



Blockquote

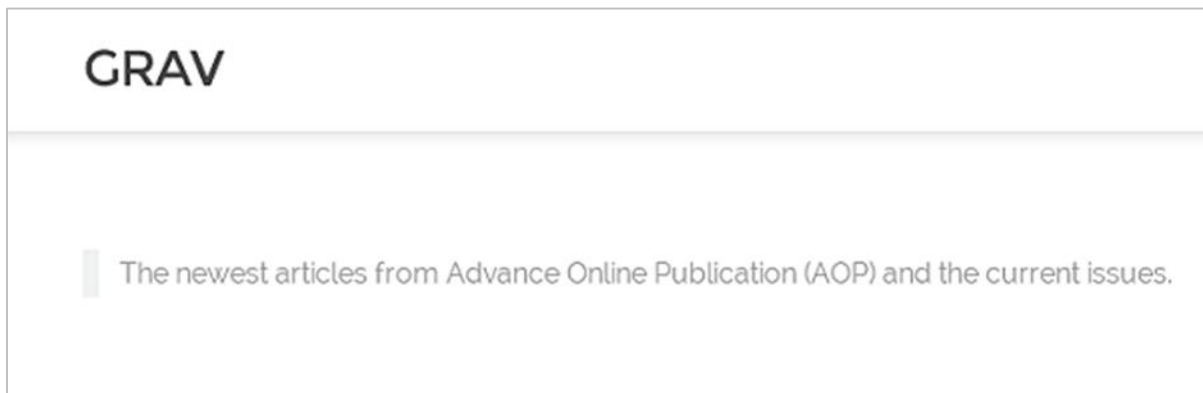
To create a block quote, you must add an `>` sign before the sentence or the word.

Example

```
>The newest articles from Advance Online Publication (AOP) and the current issues.
```

In this example, we have used a `>` sign before the sentence.

Open the `.md` file in a browser as **localhost/Grav/mypage;** you will receive the following result:



Blockquote can also be used in the following way:

```
>The newest articles from Advance Online Publication (AOP) and the current issues.
>>> The newest articles from Advance Online Publication (AOP) and the current issues.
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Notices

Notices can be used to inform or notify about something important.

There are four types of notices — yellow, red, blue and green.

Yellow

You must use the `>>>` sign before a yellow notice type that describes **!Info** or information.

Example

```
>>>Neurotransmitter-gated ion channels of the Cys-loop receptor family are
essential mediators of fast neurotransmission throughout the nervous system and
are implicated in many neurological disorders.
```

Red

Use four `>>>>` signs before a red notice for a Warning.

Example

```
>>>>Neurotransmitter-gated ion channels of the Cys-loop receptor family are essential mediators of fast neurotransmission throughout the nervous system and are implicated in many neurological disorders.
```

Blue

Use five >>>>> signs for a Blue notice type, this describes a Note.

Example

```
>>>>>Neurotransmitter-gated ion channels of the Cys-loop receptor family are essential mediators of fast neurotransmission throughout the nervous system and are implicated in many neurological disorders.
```

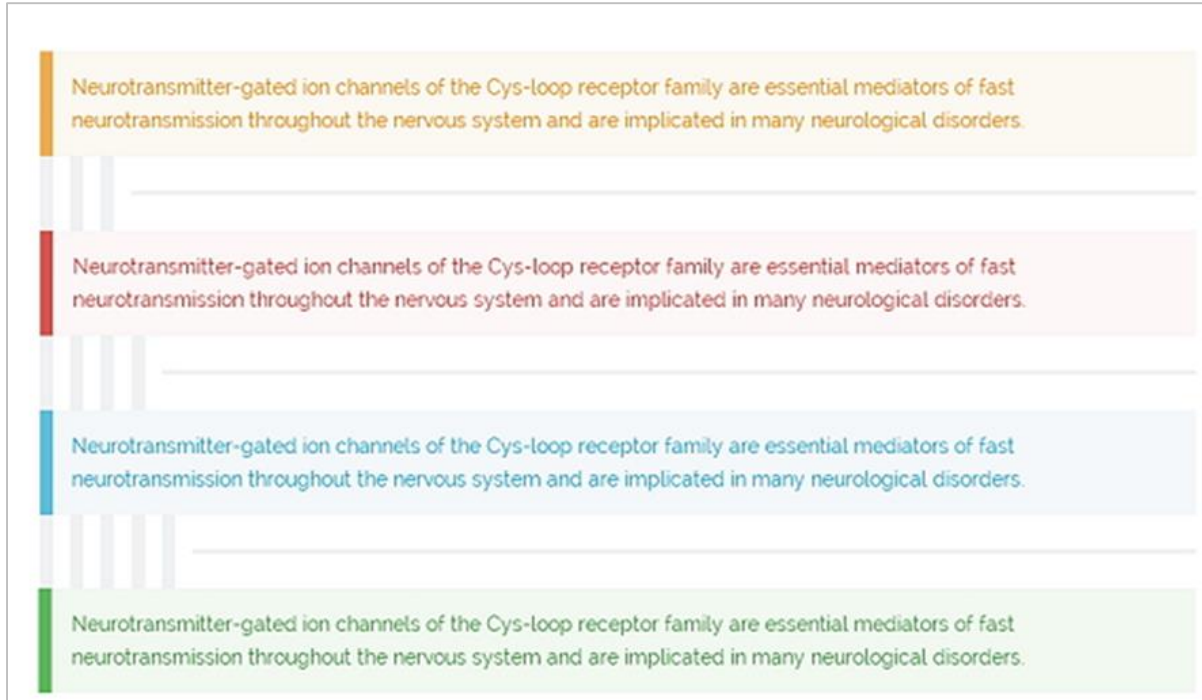
Green

Use six >>>>>> signs before a Green notice type, this describes a Tip.

Example

```
>>>>>>Neurotransmitter-gated ion channels of the Cys-loop receptor family are essential mediators of fast neurotransmission throughout the nervous system and are implicated in many neurological disorders.
```

Open the md file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Lists

In this section, we will understand how the unordered and ordered lists work in Grav.

Unordered

In an unordered list, bullets are used. Use `*`, `-`, `+` symbols for bullets. Use the symbol with space before any text and the bullet will be displayed.

Example

```
+ Bullet
+ Bullet
+ Bullet
  -Bullet
  -Bullet
  -Bullet
  *Bullet
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



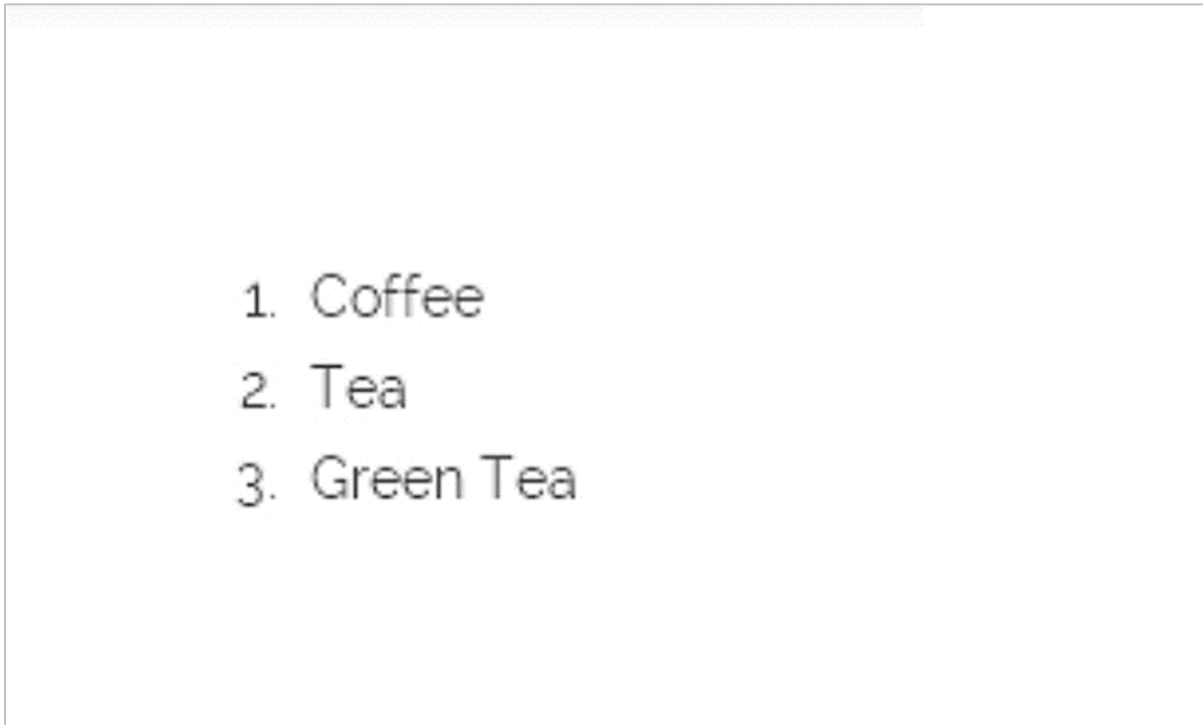
Ordered

Add the numbers before you list something.

Example

1. Coffee
2. Tea
3. Green Tea

Open the `.md` file in a browser as **localhost/Grav/mypage**. This will give you the following result:



Code

In this section, we will understand how the Inline and block code “fences” work in Grav.

Inline Code

Make inline code using (```) for using codes in markdown.

Example

In the given example, '`<section></section>`' must be converted into code.

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:

In the given example `<section></section>` must be converted into code.

Block code “fences”

Use (````) fences if you want to block multiple lines of code.`

Example

```

```
Your Text Here
```

```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Tables

In Grav, tables are created by using pipes and dashes under the header section. Pipes must not be vertically aligned.

Example

```
| Number| Points |
| -----| ----- |
| 1      | Eve Jackson 94|
| 2 | John Doe 80 |
| 3      |Adam Johnson 67 |
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:

Number	Points
1	Eve Jackson 94
2	John Doe 80
3	Adam Johnson 67

Right Aligned Text

To get the table contents at the right, you must add a colon on the right side of the dashes below headings.

```
| Number| Points |
| -----:| -----: |
| 1      | Eve Jackson 94|
| 2 | John Doe 80 |
| 3      |Adam Johnson 67 |
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:

Number	Points
1	Eve Jackson 94
2	John Doe 80
3	Adam Johnson 67

Links

In this section, we will understand how links work in Grav.

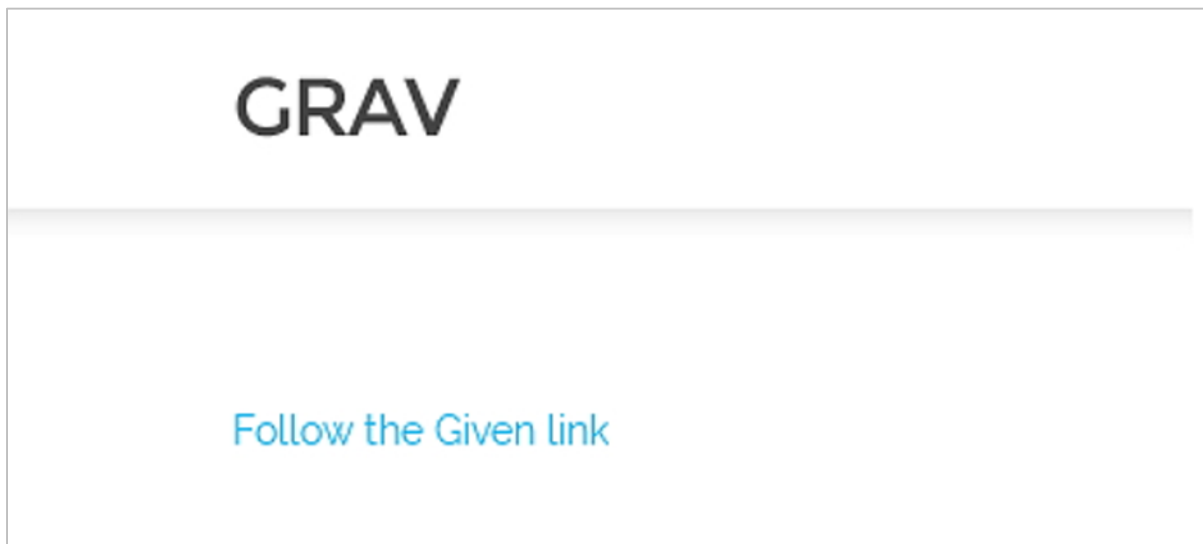
Basic Links

Links are made with the help of ([]) square brackets and (()) parenthesis. In [] brackets, you must write the content and in () write the domain name.

Example

```
[Follow the Given link](http://www.google.com)
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Add a Title

In this section, we will understand how to add a title in `.md` file.

Example

```
[Google](https://www.gogle.com/google/ "Visit Google!")
```

Open the `.md` file in a browser as **localhost/Grav/mypage**; you will receive the following result:



Images

Images are similar to a link but have an exclamation point at the start of the syntax.

Example

```
![Nature] (/Grav/images/Grav-images.jpg)
```

End of ebook preview
If you liked what you saw...
Buy it from our store @ <https://store.tutorialspoint.com>