A large blue rectangular box containing the title text in white.

The **Ultimate** Theme Guide for Designers

Vanilla Forums

Topics Covered

[Introduction](#)

[Tool Box](#)

[Choosing Your Theme](#)

[Homepage Layout](#)

[Homepage](#)

[Layouts](#)

[Customize HTML](#)

[Basic HTML layout](#)

[Understanding HTML Layout](#)

[Breaking down and customizing the code](#)

[The HTML head element](#)

[The HTML body element](#)

[The Head](#)

[The Body](#)

[Footer](#)

[Understanding Template Tags](#)

[Customizing CSS](#)

[Move the location of body and panel](#)

[Modifications in Bootstrap theme when you add a logo](#)

[Remove Gender from the registration form](#)

[Q & A Plugin](#)

[Ranks Plugin](#)

[Modifying CSS for Roles](#)

[Displaying Icons For Roles and Ranks](#)

[Add CSS for Categories](#)

[Pockets](#)

[Getting Started With Pockets](#)

[Adding A Pocket](#)

[Advance Techniques with Pockets](#)

[Conclusion](#)

[Appendix](#)



Introduction

Congratulations on your new community! We are excited to have you as a new customer. The next step is to make your community look great. We have created the following guide to help you customize the look and feel of your forum. We hope you find it helpful.

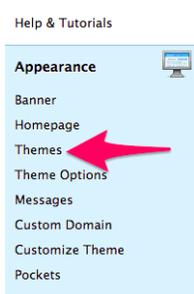
Tool Box

To assist you in modifying the base theme, you will want to use either Google Chrome or Mozilla Firefox that have developer tools that let you inspect the site CSS and HTML. You can also consider free add-ons such as Firebug.

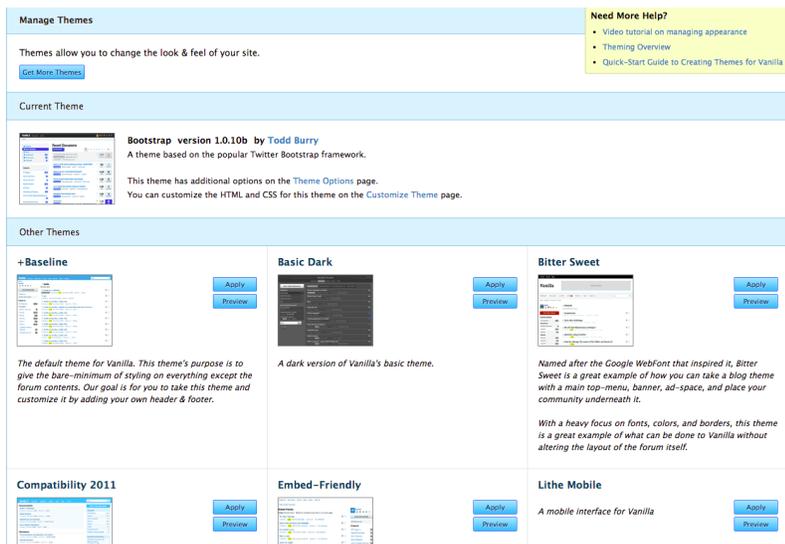
Learn more about Google Chrome Dev Tools [here](#) and Mozilla Firefox Dev Tools [here](#).

Choosing Your Theme

Your first step is to choose a theme or stay with the default theme. You can change your theme under the Appearance section of the Dashboard.



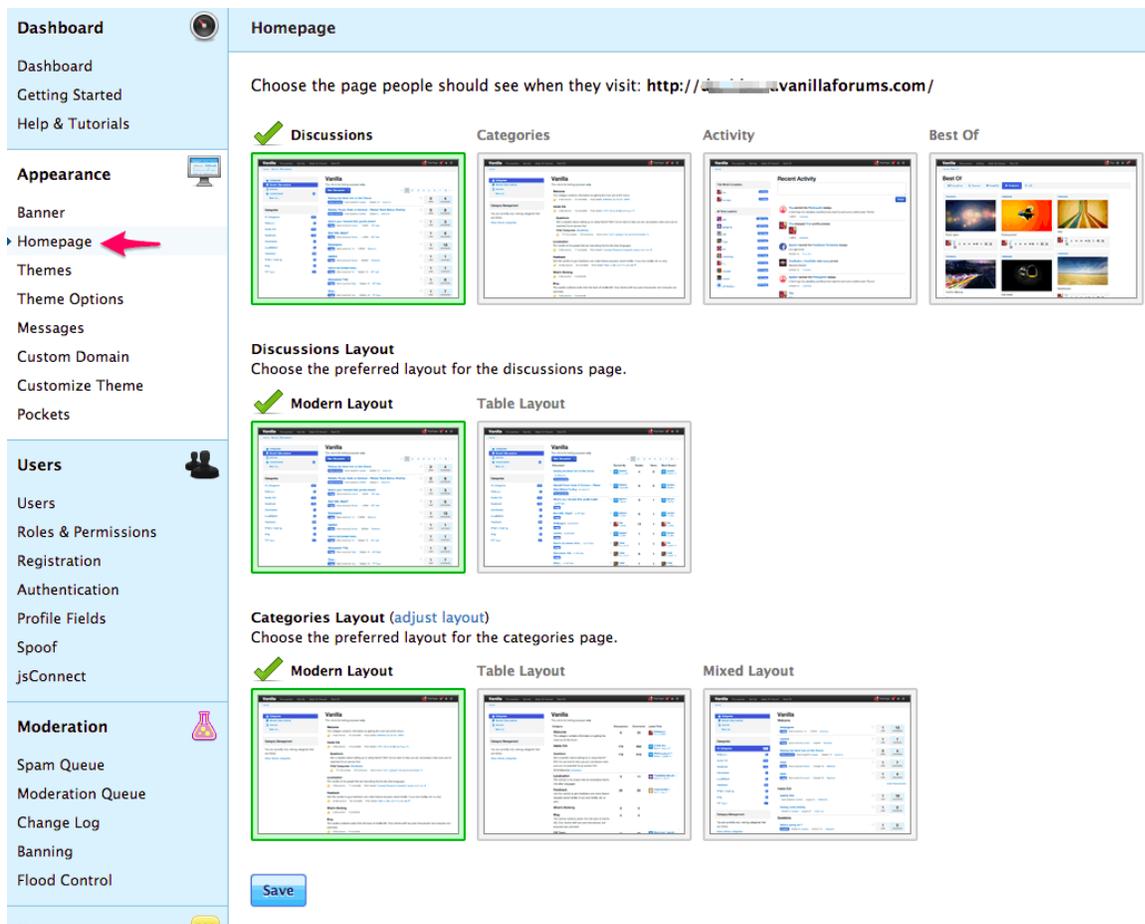
Once clicked you will see the “Manage Themes” page



Think of the theme you are choosing as a base canvas. As you make changes to the HTML or CSS, it will override this base canvas.

Homepage Layout

Once you have settled on a layout, and before you start modifying your HTML you will want to consider selecting a homepage layout, as this might impact your design.



You have three things to consider when setting the home page and layout:

1. **Homepage:** what will people see when they land on your forum?
2. **Discussion layout:** how will discussions look?
3. **Category layout:** What kind of page layout will categories take?

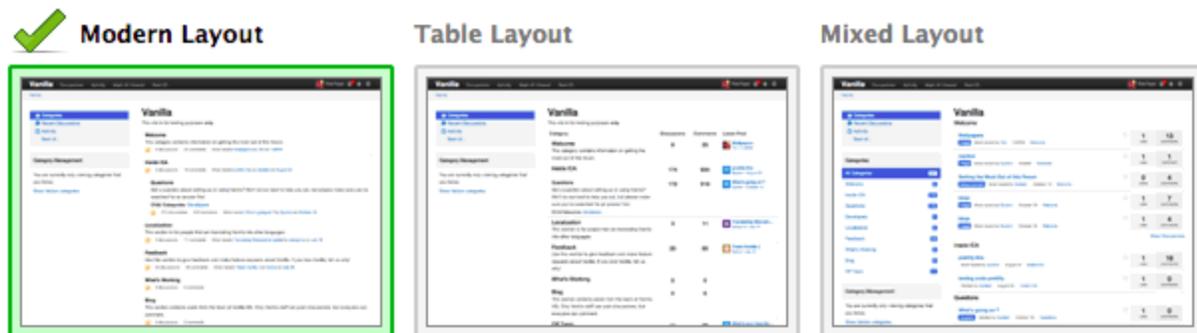
Homepage

The vast majority of forums will display either the recent discussions across all categories or a list of categories as a home page. If you have migrated your large community to Vanilla and things were divided by categories, you might want to continue this practice. The “Discussions” view allows your visitors to view most recent discussions, regardless of category. This is great for newer communities with less content and it gives members a quick view of what’s new.

There is the activity view, which turns your forum more into a social network, and just shows off the activity in your community. Almost like a mini social network.

Finally, there is the Best of view. This is a page that will begin to populate as you get content on your site, so it may not appear to start. Note: this page does not populate if you are not using the Reactions feature.

Layouts

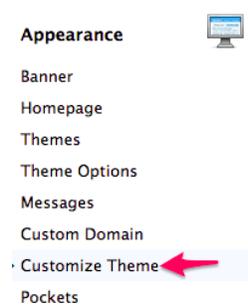


Next is the layout for the discussion list and category pages, the table layout is something your users might be familiar with, so it’s a great choice when migrating from another platform. We also have a more modern layout, which is our original take on the old style table-based layout. Once you have settled on the layout it’s time to customize the template. Note: You can always change your layout and home page selections at a later time if you

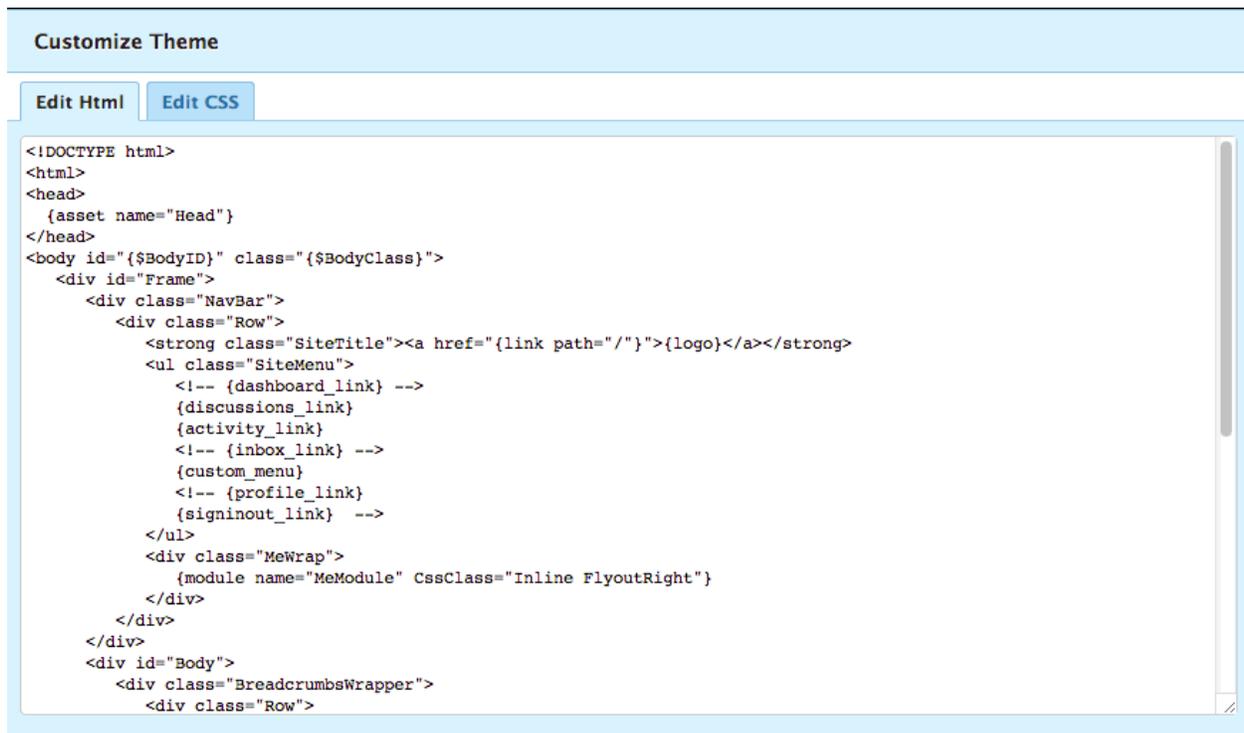
choose. This might require some CSS changes.

Customize HTML

Now that you have all the other stuff out of the way, it's now time to really have fun and start customizing some HTML. Once you are ready, click the "Customize Theme" link under the appearance menu in the Vanilla Forums dashboard.



This will open Vanilla's version of a theme editor with two tabs. One for HTML and one for CSS (we will focus on the CSS later).

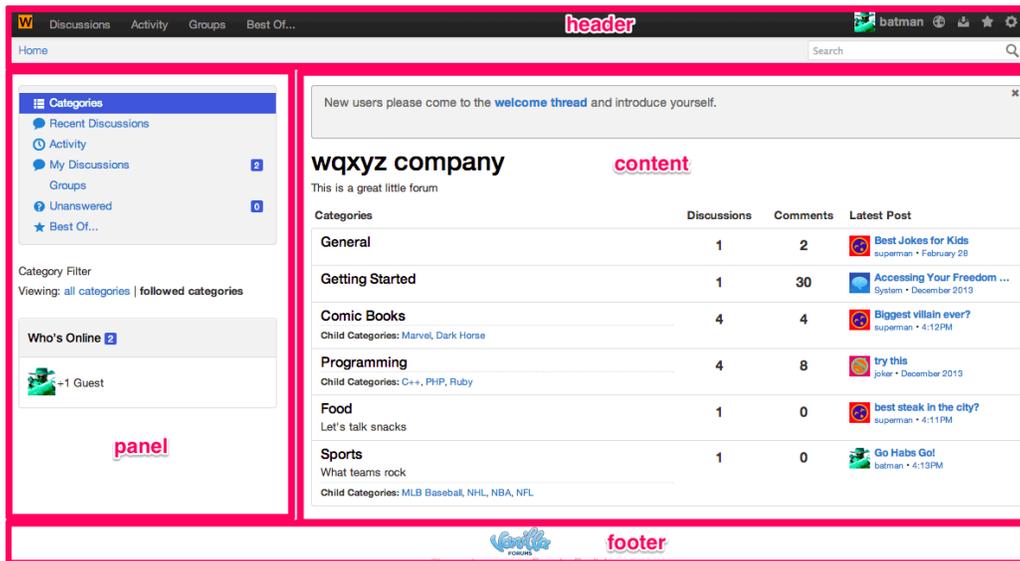


In our editor we offer the ability to preview changes before they go live, and label revisions. This makes it easy to rollback you theme to an earlier state if you choose. You can find revision options.



Basic HTML layout

The easiest way to decipher the HTML for your forum is to break it down into parts: Header, which includes the menu and your logo, Content, Panel, which is your sidebar, and Footer (see diagram below)



Understanding HTML Layout

Breaking down and customizing the code

To help make the code easier to understand, we will break down the code into sections starting from the top.

The HTML head element

```
<head>  
{asset name='Head'}  
</head>
```

The head element is a container for other head specific elements. Elements inside can include scripts, instruct the browser where to find style sheets, provide meta information and more. The {asset name='Head'} tag adds the following to the head section: <base>, <link>, <meta>, <script>, <style>, and <title>.

*NOTE: DO NOT REMOVE this tag. There is no need to add any other elements to the head in order for your forum to function or to customize your forum.

The HTML body element

The next line of code contains 2 variables

```
<body id="{ $BodyID}" class="{ $BodyClass}">
```

These 2 variables enable both the default CSS (Cascading Style Sheet) and custom CSS to properly style the look and feel of your forum – We will cover the Edit Css tab in Part 2 of this tutorial.

*NOTE: DO NOT REMOVE this tag.

The Head

Not to be confused with `<head></head>`. `<div id="Head">` is a unique identifier (id) to an element (div) that styles the header of your forum, which contains your logo and menu. In some themes, (Like Bootstrap) this might be represented by `<div id="Frame">` .

The Body

Not to be confused with `<body></body>`. `<div id="Body">` is a unique identifier (id) to an element (div) that styles the body (content) of your forum, which contains your discussions and panel (sidebar).

There are 2 important elements here: `<div id="Content">` and `<div id="Panel">`.

The `<div id="Content">` is wrapped around the `{asset name="Content"}` tag which displays your helper menu and discussion list. The `<div id="Content">` controls the look and feel (can be modified in the CSS).

The `<div id="Panel">` is wrapped around the `{asset name="Panel"}` tag and depending on the theme includes the start a new discussion, categories list, bookmarked discussions and more. Just like with the `<div id="Content">` the `<div id="Panel">` controls the look and feel (can be modified in the CSS).

Footer

Finally, let's take a look at the Foot. `<div id="Foot">` is a unique identifier (id) to an element (div) that styles the footer of your forum, which contains the "Powered by Vanilla" link.

If you are looking to learn more about HTML tags, check out [w3Schools](http://w3schools.com).

Understanding Template Tags

A template tag is a code that instructs Vanilla to “do” or “get” something. Our template system uses a modified smarty system. Some template tags can have values added to them. For example, the asset tag displays blocks of information, like your discussions or the side panel.

```
{asset name="Content"}  
{asset name="Panel"}
```

The following are some of the current tags available for use as you theme:

asset: The asset tag is used to display blocks of information, like your discussions or the side panel. This tag must have a value associated with it in order to function correctly.

Example Usage:

```
{asset name="$value"}
```

Values

- Head – Displays Head data between the <head></head> HTML tags.
- Content – Displays Discussion data.
- Panel – Displays Panel data. For example; Categories, Bookmarked Discussions, Start New Discussion Button.
- Foot – Loads functions hidden scripts tags and other custom code in order for Vanilla to function correctly.

link: The link tag is used to simplify linking to different views in your forum. For example you can write {link path= “discussions”} inside the href HTML tag instead of the full path. This tag must have a value associated with it in order function correctly

Values

- dashboard/settings – links to your dashboard.

- discussions – links to discussions, the default landing page for your forum.
- activity – links to the activity field.
- messages/inbox – links to your inbox.
- profile – links to your profile page.
- signout – will toggle between sign in and sign out.

Example Usage:

The below examples will create a link to your dashboard inside a list element.

```
<li><a href="{link path="dashboard/settings"}">Dashboard</a></li>
```

or to an external link like your blog

```
<a href="{link path="http://www.somesite.com/blog/"}">Blog</a>
```

literal: The literal tag is used to enclose content that you don't want to be parsed by the custom theme. For example you can use this to add Google Analytics javascript code.

Example Usage:

```
{literal}
<script type="text/javascript">
var _gaq = _gaq || [];
_gaq.push(['_setAccount', 'UA-xxxxxxx-1']);
_gaq.push(['_setDomainName', 'somesite.com']);
.....
</script>
{/literal}
```

logo: The logo tag is used to display a custom logo which can be uploaded under banner in the dashboard.

Example Usage: This will display your custom logo inside an H1 tag, or banner if an image is present.

```
<h1>{logo}</h1>
```

searchbox : The searchbox tag is used to display the default search function.

Example usage:

```
<div id="Search">{searchbox}</div>
```

If you are using Advanced Search, you will want to use: {searchbox_advanced} in your theme. BUT this can only be used after the Advanced Search plugin has been enabled.

breadcrumbs:The breadcrumbs tag is used to display navigation breadcrumbs

Example Usage:

```
{breadcrumbs}
```

if : This is a conditional tag . This tag must have a value associated with it in order to function correctly. You can find the full documentation for the {if} tag at:

<http://www.smarty.net/docsv2/en/language.function.if.tpl>

Example Usage:

```
{if $value}
<li><a href="{link path="speciallink/moderator"}">Special Link</a></li>
{/if}
```

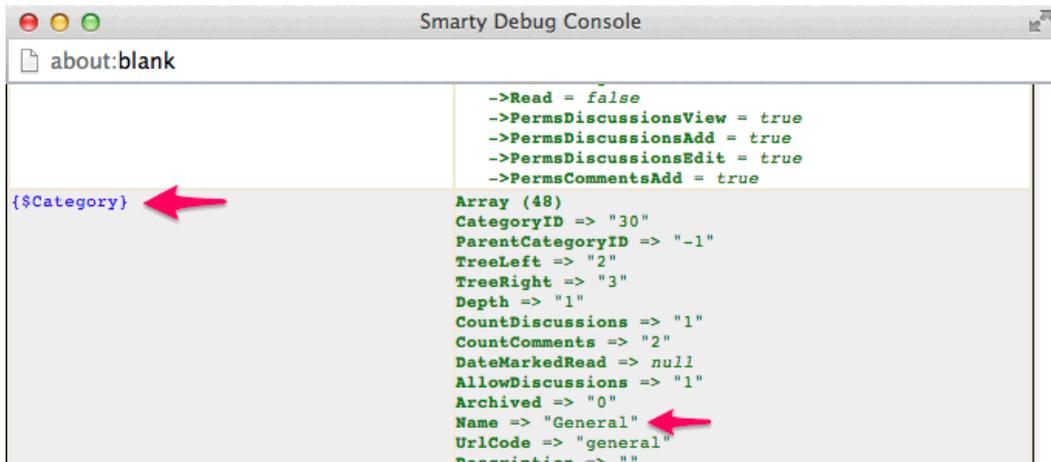
or an if else conditional

```
{if $value}
<li><a href="{link path="speciallink/moderator"}">Special Link</a></li>
{else if $value}
<li><a href="{link path="speciallink/guest"}">Nothing Special</a></li>
{else}
<li><a href="{link path="speciallink/member"}">Special Link 2</a></li>
{/if}
```

Values (here are only a few \$value examples to get you started):

- Content viewable by people with Manage Roles:
CheckPermission('Garden.Settings.Manage')
- Content viewable by people with Admin Roles:
CheckPermission('Garden.AdminUser.Only')
- User is signed in : \$User.SignedIn
- If homepage is set to Categories show content:InSection("CategoryList")
- If category named “general” show content: \$Category.Name == "General"
- If in any of the sections defined in the array: {InSection(array([sections]))}

You can also see all of the data available to you with the {debug} tag. This will launch a popup so you may have to unblock it in your browser. (Following example shows how you get the syntax for category)



```

Smarty Debug Console
about:blank

->Read = false
->PermsDiscussionsView = true
->PermsDiscussionsAdd = true
->PermsDiscussionsEdit = true
->PermsCommentsAdd = true

Array (48)
CategoryID => "30"
ParentCategoryID => "-1"
TreeLeft => "2"
TreeRight => "3"
Depth => "1"
CountDiscussions => "1"
CountComments => "2"
DateMarkedRead => null
AllowDiscussions => "1"
Archived => "0"
Name => "General"
UrlCode => "general"
Description => ""
  
```

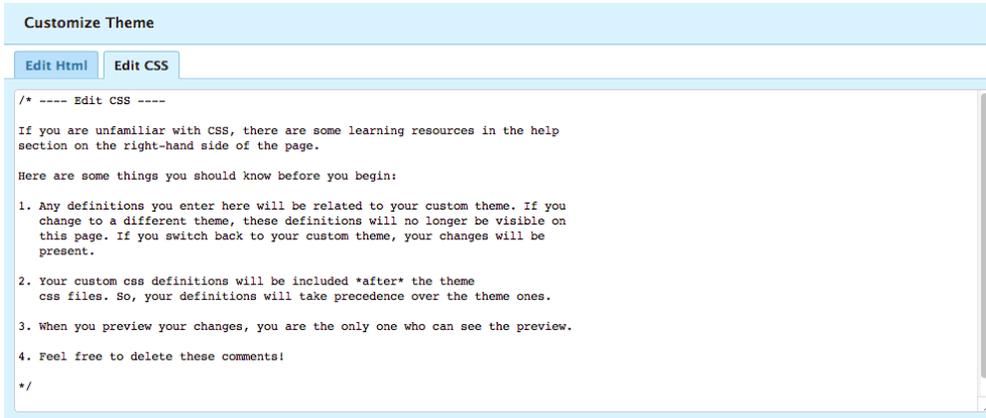
debug:

Smarty has a built in debug mode which Vanilla leverages and is useful for developing themes with. Place {debug} in the section in the HTML it will trigger Smarty's built in variable listing. This will show you all variables set at the current scope. You can use CTRL +F to search the data.

*Note that this template tag should never be left enabled in a production environment. A quick way is to comment it out with smarty *, so for example {*debug*}.

Customizing CSS

Hand in hand with customizing HTML, comes modifying CSS. Vanilla offers you the flexibility to add whatever CSS you want and really customize the look and feel of your community. To access the CSS editor, you just click “customize theme” in the appearance sidebar. Once there, click the “CSS” tab and you will be presented with the following screen:



Like with customizing HTML, you are able to get a preview of your changes and you are also able to roll back revisions.

While we cannot give you a course on CSS, there are plenty of great CSS resources on the web. Using the browser development tools will be invaluable in customizing your theme, but to help you along, below are some common CSS tweaks you can make:

Move the location of body and panel

The below will help you move the sidebar from left to right.

```
/* Move the sidebar from the right to the left side */
```

```
#Content {  
    float: right;  
}  
  
#Panel {  
    float: left;  
}
```

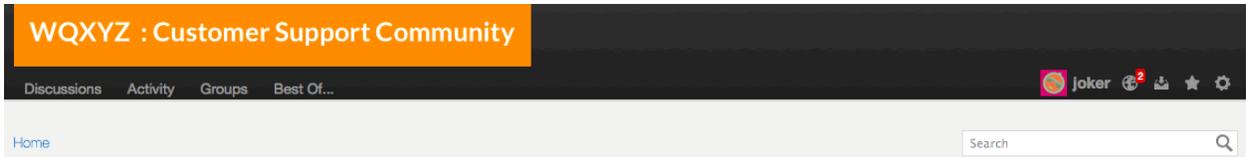
You might need to do some other tweaks depending on the theme. Below is an example on how to move the panel in the Bootstrap theme.

```
#Panel {
  float: right;
  width: 300px;
}

#Body .ContentColumn {
  margin: 0 320px 0 0;
}

#Body .ContentColumn {
  margin-left: 0 !important;
}
```

Modifications in Bootstrap theme when you add a logo



If you add a Logo image in the Bootstrap theme, here is how to move links and show search with Bootstrap theme. You will need to make navbar bigger so search will show too.

*/*Bootstrap banner & search fix */*

```
div.NavBar {
  height:90px;
}

div#Frame {
  margin-top:80px;
}

div.MeWrap {
  margin-top:-10px;
}
```

To move the menu bar over, you will need to close the DIV to (so a bit of HTML)

Change:

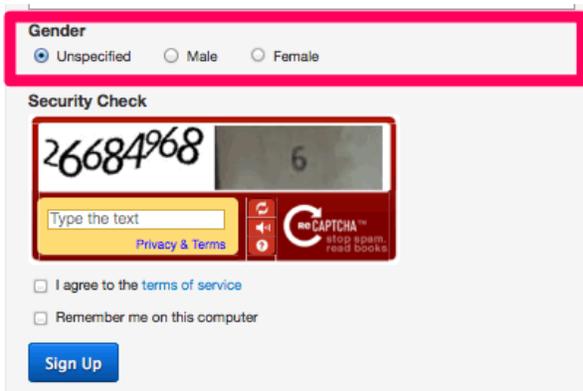
```
<div class="Row">  
  <strong class="SiteTitle"><a href="{link path="/"}">{logo}</a></strong>
```

to this

```
<div class="Row">  
  <strong class="SiteTitle"><a href="{link path="/"}">{logo}</a></strong>  
</div>  
<div class="Row">
```

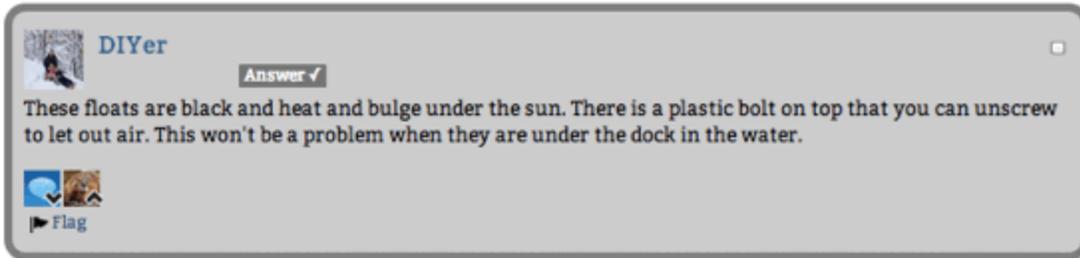
Remove Gender from the registration form

On the registration page, for new users, you may want to hide the gender question:



```
/* hide gender on registration */  
li.Gender {  
  display:none;  
}
```

Q & A Plugin



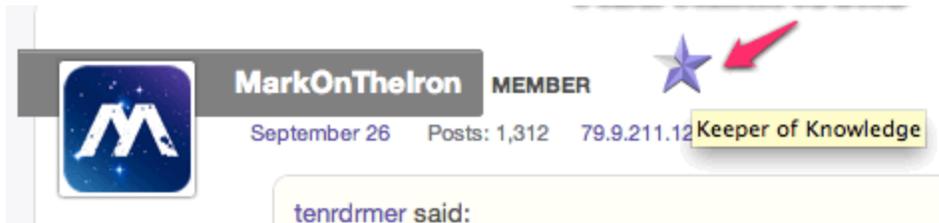
You can modify the CSS for accepted answers if you are using the Q & A plugin.

```

/* Accepted Answers */
ul.AcceptedAnswers {
  border: solid thick grey;
  background-color: #CCC;
  border-radius: 15 px;
  -webkit-border-radius: 15px;
  -moz-border-radius: 15px;
}

```

Ranks Plugin



You can enter a CSS class to style anything in the comment block (comment body and profile info). Once you turn on Ranks, the CSS class to differentiate each rank is added automatically. As you can see the following example the Admin, is given a class called "Rank-Admin".

if you want to modify the looks of posts by users of certain Ranks add a custom CSS rule to your "Edit CSS" tab in Customize Theme. For example, to change the background of comments we can add something like:

```
li.ItemComment.Rank-Admin {
background-color: #F5F8E8;
}
```

For both discussions and comments use:

```
.ItemDiscussion, li.ItemComment.Rank-Admin {
background-color: #F5F8E8;
}
```

Depending on the theme, the below might work as well:

```
Item.Rank-Admin {
background-color: #F5F8E8;
}
```

If you plan to add customization to other Ranks including custom Ranks, make sure you edit the specific Rank and give it an appropriate CSS class. In the following example, a new CSS class called “Rank-Newb” is added to a Rank called ‘Level 1’. This keeps the naming convention consistent and ensure no conflicts in the existing theme.

Edit Rank

Name

Label
This label will display beside the user. It can be the same as the rank's name or have a more visual appearance. HTML is allowed.

Level
The level of the rank determines it's sort order. Users will always be given the highest level that they qualify for.

CssClass
You can enter a css class here and it will be added to certain elements on the page. You can combine this with custom theming to add some great effects to your community.

Add a css class to make customization easier



In the above examples we showed you how to change the background of specific Roles and Ranks, but this is only the beginning. In this next section, we want to show you how you can have a bit more fun in the customization of Roles and Ranks.

Modifying CSS for Roles

If you are not using Ranks, you will want to enable the “Roles Titles” plugin. This adds a role specific CSS class for you to target. In the following example, we can see, using Firebug (but use whatever dev tool you like), that a new class “Role_Administrator” is added for users of the administrator account.

The next step is to add a custom CSS rule to your “Edit CSS” tab in Customize Theme. So for example to change the background of comments we can add something like:

```
li.ItemComment.Role_Administrator {
background-color: #F5F8E8;
}
```

If you want to change colour of the discussions started by the Administrator role use:

```
.Item.ItemDiscussion.Role_Administrator {
background-color: #F5F8E8;
}
```

If you want all content (comments and discussions) of an admin to be highlighted use:

```
.Item.Role_Administrator {
background-color: #F5F8E8;
}
```

Finally, remember to change “Role_Administrator” in the above example to the appropriate role as needed. i.e Moderator is Role_Moderator. We have used the example of changing the background, but you can apply any CSS changes you want.

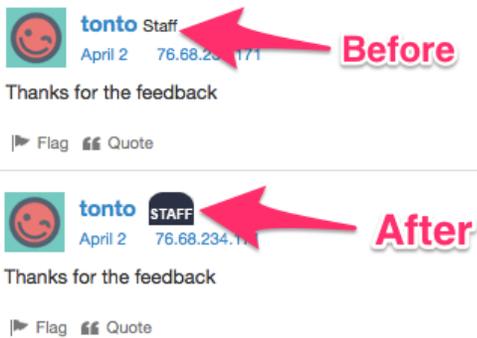
Displaying Icons For Roles and Ranks

There are some other ways you can use CSS and images to differentiate Roles and Ranks. Here are some other examples to get your creative juices flowing.

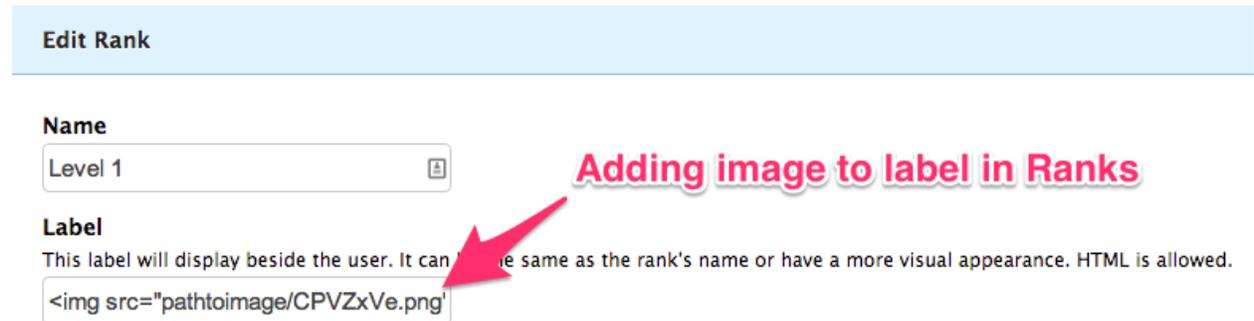
If you wanted to add a “Staff” icon to a Staff Role, for example, you can do that with CSS. Using the “Role Titles” plugin, we get the “Role_Staff” CSS class. By using the following CSS:

```
.Item.Role_Staff .MItem.RoleTitle {
color: transparent;
background: no-repeat url(/pathtoimage/staff.jpg);
content: " " !important;
position: absolute;
width: 30px;
margin-left: 8px;
}
```

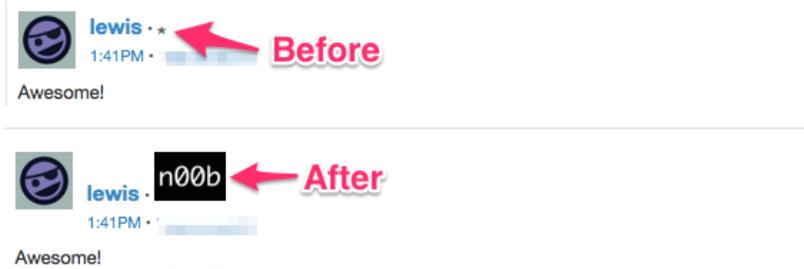
We can have an end result like this:



Depending on the theme you are using, you may have to modify the CSS to fit your needs. For Ranks, it’s a bit different. If you decide to add an image icon to a specific Rank, no custom CSS changes are needed. To add an image icon in Ranks, the way to do this is to add some HTML in the “Label” form field when editing a specific Ranks’ detail. For example, our Rank Newb, example from before, you would add HTML to the label section.



The end result:



As you can see there is a chance to really get creative and give your community a very unique look and feel. Please note the above examples were done using the “Bootstrap theme”, but the concepts are the same. *For best results, make sure to use a proper tools like Google Chrome Dev Tool or Firebug to identify classes to modify.*

Add CSS for Categories

When you create a new category, you have the option to add a class.

Add Category

Categories allow you to organize your discussions.

Category

Category Url: <http://vanillaforums.com/categories/premium-content/> [edit](#)

Description

Css Class

Adding a custom class to a category

By adding this custom class you can then use CSS to alter its appearance.

```
/*CSS for Premium Category*/
tr.Item.Category-premium-content.Item-premium-content
{
background-color: #71ff10;
}
```

Here is the end result

Categories	Discussions
Premium Content This is a category with premium content	55
General	1030
Getting Started	756

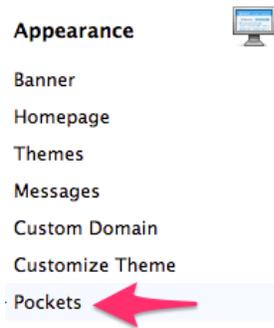
Pockets

Pockets are placeholder slots available throughout your forum. They are a great way to add little bits of HTML or Javascript to your site. If you don't have Pockets enabled, and you are on the Advanced plan (or higher), please send a request to support to have it turned on.

Word of caution: Pockets are extremely powerful, so they should be used with caution. The wrong code can break your forum, so always make sure to try things first in "Test Mode".

Getting Started With Pockets

Once you have pockets enabled, you will have a new menu item in your dashboard:



On the Pocket settings page, you will see the following:

Pockets allow you to add free-form HTML to various places around the application.

Show Pocket Locations	This option shows/hides the locations where pockets can go, but only for users that have permission to add/edit pockets. Try showing the locations and then visit your site.
Add Pocket	Add a new Pocket to your site.

You can turn on “Show Pocket Locations” to see all the slots available on your forum. Only Administrators can see them, so it’s a great way to see where you want to add a pocket. The following image gives you an idea.

HEAD (1)

HEAD (AFTER)

Home

PANEL (BEFORE)

- Categories
- Recent Discussions**
- Activity
- My Discussions **5**
- My Drafts **5**
- Groups
- Unanswered **0**
- Best Of...

PANEL (1)

Categories

All Categories	4
COMIC BOOKS	
DC Comics	1
Dark Horse	0
Marvel	0
PROGRAMMING	
Ruby	0
GENERAL	

PANEL (2)

Who's Online **1**

PANEL (3)

PANEL (AFTER)

CONTENT (BEFORE)

This is some HTML

Recent Discussions

New Discussion -

Batman is the best ☆ **1** **0**
System Started by System November 25 DC Comics
view comments

BETWEEN DISCUSSIONS (1)

Undefined discussion subject. ☆ **1** **1**
System Most recent by System November 1 General
view comment

BETWEEN DISCUSSIONS (2)

Testing New Discussion ☆ **21** **7**
Announcement batman Most recent by System October 25 General
views comments

BETWEEN DISCUSSIONS (3)

Vanilla Forums - Testing new Discussion ☆ **21** **0**
David Started by David October 23 via Email General
views comments

New Discussion -

CONTENT (AFTER)

FOOT (BEFORE)

FOOT (1)

FOOT (AFTER)

Adding A Pocket

Now you are ready to add a pocket. Click the “Add Pocket” button and you will be presented with the following page to create your pocket:

Name
Enter a descriptive name for the pocket. This name will not show up anywhere except when managing your pockets here so it is only used to help you remember the pocket.

 Name to give to your pocket

Body
Enter the text of the pocket. This will be output exactly as you type it so make sure that you enter valid HTML.

This is where you add HTML content

Page
Select the location of the pocket.
(All) ▾ **What page of the forum you want the pocket to appear**

Location
Select the location of the pocket.
(Select a Location) ▾ **What section of the page you wish to appear**

Repeat

- Before
- After
- Repeat Every
- Given Indexes

Conditions

Limit the display of this pocket to "mobile only".

- Only display on mobile browsers.
- Limit the display of this pocket for mobile devices.
- Never display on mobile browsers.

Limit the display of this pocket for embedded comments.

- Don't display for embedded comments.

Most pockets shouldn't be displayed in the dashboard.

- Display in dashboard. (not recommended)

Enable/Disable

- Enabled: The pocket will be displayed.
- Disabled: The pocket will **not** be displayed.
- Test Mode: The pocket will only be displayed for pocket administrators.

New pockets should be added in test mode to make sure they work as you wish.

Here is some information about each part of the “Add Pocket” form:

Name: This is the name for your pocket. Give it a descriptive name to distinguish different pockets.

Body: Here is where you can add snippets of HTML or Javascript. This is where you could add your Adsense, Doubleclick or any Ad Network you want.

Page: Use this to control which page your pocket will appear on.

Location: This is the location of the actual pocket. So for example, discussion page or the footer.

Repeat: This is how you can decide to have your ad show every third comment in a discussion or just have an ad just before the footer.

Conditions: With conditions, you control pocket behaviour. For example, you can create a pocket with an ad that will only be seen on mobile devices.

Enable/Disable: You should always enable the pocket in “Test Mode” until you are happy with the way it looks. When a pocket is in test mode, only those with Pocket Admin rights will be able to see your content.

Advance Techniques with Pockets

Pockets also adds another level of customization to your site and the HTML template too. You can add a Pocket to your template directly.

For example you can create a Pocket called “Foo” and in the body add html or JS.

Add Pocket

Name
Enter a descriptive name for the pocket. This name will not show up anywhere except when managing your pockets here so it is only used to help you remember the pocket.

Body
Enter the text of the pocket. This will be output exactly as you type it so make sure that you enter valid HTML.

some html or js

The other settings are not really going to matter, as you will see in a moment. Also unlike how you normally would enable a pocket, you will select “disabled”

Enable/Disable

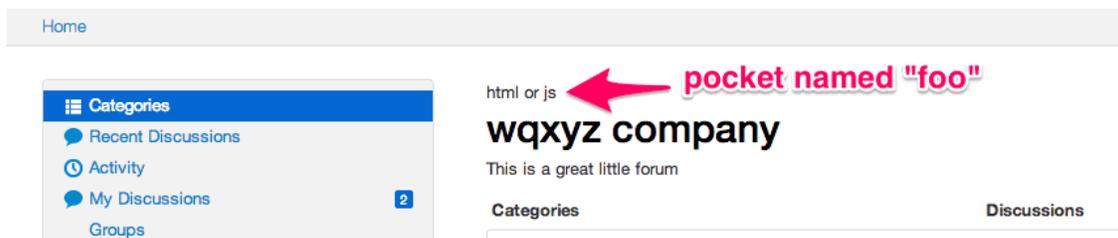
- Enabled: The pocket will be displayed.
- Disabled: The pocket will **not** be displayed.
- Test Mode: The pocket will only be displayed for pocket administrators.

The final step is add the pocket to your HTML template where you want the pocket to appear. In the below example, we have added `{pocket name="foo"}` to be just above content container.

```

Edit Html Edit CSS
<div id="Body">
  <div class="BreadcrumbsWrapper">
    <div class="Row">
      <div class="SiteSearch">{searchbox}</div>
      {breadcrumbs}
    </div>
  </div>
  <div class="Row">
    <div class="Column PanelColumn" id="Panel">
      {asset name="Panel"}
    </div>
    <div class="Column ContentColumn" id="Content">
      {pocket name="foo"}
      {asset name="Content"}</div>
    </div>
  </div>
</div>
    
```

And voila!



Another great aspect of this, is you can use smarty conditional tags as we showed earlier. So for example to show “foo” on the index page if the Homepage is set to Categories, you could do it as so:

```

{if InSection("CategoryList")}
  {pocket name="foo"}
{/if}
    
```

If you wanted to show “foo” only in the general category, the syntax would be:

```

{if $Category.Name == "General"}
  {pocket name="foo"}
{/if}
    
```

Adding a conditional to user who is SignedIn

```
{if $User.SignedIn}
{pocket name="foo"}
{/if}
```

Adding a conditional to user who is not SignedIn

```
{if !$User.SignedIn}
{pocket name="foo"}
{/if}
```

Conclusion

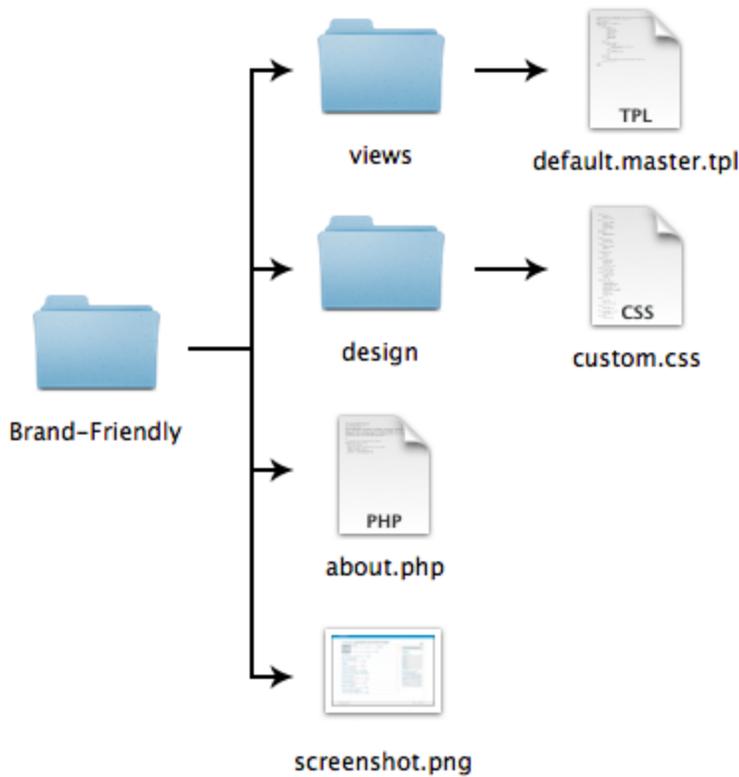
There are lots of creative ways to design a Vanilla theme and we hope this guide will help you in creating an awesome forum theme for your community. Just remember, we could not show you everything, but we think this guide should get you on your way...

If you still feel like this is too hard, or if you are not comfortable in doing it yourself, you can contact our sales team for a quote on a custom theme.

Appendix

ATTENTION: The following information is for special VIP customers wanting to theme using our open source project. **If you are not a VIP customer, please note, files you create will not be uploaded to our server.** To get those changes to a hosted theme copy and paste to the respective HTML and CSS tabs in the customize theme editor in your Vanilla dashboard. Also, the below information assumes you are comfortable in using a local environment.

If you are ready to go, grab the latest version of [Vanilla from Github](#) (MASTER), so you can customize a theme in your local environment . A theme will be composed of the following elements:



Make a copy of the default theme in a new folder, open the about.php and change the values so you will be able to identify it as your custom theme. The following is an explanation of the other elements of a theme to assist you as you work on it locally.

Elements of a Theme

[about.php](#) - This file contains information about your theme, such as Theme Name, Theme Description, Version, Author, Author Email, and AuthorUrl. This file is required to register your theme in Vanilla so that it can be enabled via the Appearance/Themes page in the Dashboard.

[screenshot.png](#) - A thumbnail image of your theme used to help display your theme in Appearance/Themes page in the Dashboard.

[Views Folder](#) - This folder contains the default.master.tpl – the master view for your theme.

[default.master.tpl](#) - This file is the master view for your theme. It contains the html for the header and footer of every page. All other views – i.e. discussions, activity, profile, and inbox – are displayed *within* this one. So, you can use this file for customizing your banner, logo, footer, etc. **Modifications to this file will be what you will need to copy and paste over to your hosted forum in the HTML tab.**

[Design Folder](#) - This folder contains all files associated with your theme's design, such as custom.css and all image files. **We cannot import your image files. You can link to them in your CSS or upload them once you are working on the forum.**

[custom.css](#) - This file is added *after* the base theme's css, which is contained within the Vanilla application. This stylesheet will overrule all css rules defined in the base theme. **Modifications to this file will be what you will need to copy and paste over to your hosted forum in the CSS tab.**

*****Once again, please note none of your files will be uploaded to our server. Please copy and paste your modifications into the UI in Vanilla Forums Hosted platform.*****