About

There are a lot of references here - some are basic, some advanced. You may need to read some or all of the references, but please take the time to read this page before you start. We hope it will save you time in the long run. As you learn, we invite you to come back and update this page - share what you have learned.

1. Join the tikiwiki-devel mailing lists at https://sourceforge.net/mail/?group_id=64258 and ask lots of questions.
2. Join the #tikiwiki group on IRC and ask lots of questions. Read ConnectingToIrc for instructions.
3. Use Tiki and understand how it works before you start trying to change it. Set up a site, use and administer it on an everyday basis until you can fix the problems you encounter.
4. Set up your development system to use one or more of the SVN branches (there are several). The bleeding edge is trunk.
5. If Windows will be your development operating system:
   1. If you are not sure how to set up a local server and want to use IIS on a Windows machine read Windows IIS Install or Installation.
   2. There are some good "AMP" stacks available, such as XAMPP.
6. Set up an Internet available site. This will be your playground and showcase to show off your latest feature. Then let everyone on #tikiwiki know you have something new. If you haven't already been asked to join the group - you will be now.
7. SVN is one of the more complicated things you'll need to learn about Tiki Development.
   1. Read the SourceForge pages for beginners on SVN http://apps.sourceforge.net/trac/sourceforge/wiki/Subversion. It has links to a lot of references and ideas on SVN use.
   2. Ask lots of questions on IRC. Using SVN correctly could be a degree all by itself. 😏
   3. Read where to commit.
8. Read the 3 Rules.
9. If this is the first open-source project you have worked on, understand that working on an open-source project is NOT like any other development project you have ever done. You will be dealing with a large number of people who are scattered all over this beautiful world of ours. Many of our friends learned English as a second (or third) language. Misunderstandings WILL happen! Assume positive intent and practice forgiveness. All of this will require an added effort on your part. Please read KinderCode often. A good place to learn about the differences of open source development is Eric Raymond's Cathedral and the Bazaar.
10. Before you DO anything - ask for help on the Development list or on #tikiwiki. It doesn't make a lot of sense for you to spend a lot of time trying to accomplish something the Hard Way when a few questions could save you a lot of effort. Again - ask questions and use the expertise that is available.
11. If you have an idea for a new feature, suggest it on the dev list or #tikiwiki. Get some feedback and opinions on the need for the feature and ideas on how to implement it. Some in the Tiki community
have been here for 2-3 years and know the history. As Newton before you, take an opportunity to see further by standing on the shoulders of giants.

12. If you need help with something - **ASK!** Most developers are more than happy to assist when they can - but we need to know when (and where) the help is needed. This is a **very** friendly group who believe in helping each other!

13. Read the [Hello World](#) introduction in order to learn the basic structure of the Tiki environment. This article starts with a basic "hello world" page and goes on to cover permissions, menus, queries, wiki parsing, themes and more.

14. **How to get commit access**

15. Tiki is licensed under the [LGPL license](#). Take the time to understand the meaning of this license before you commit any code. By submitting to the project you are committing to that license model. Further, any code you copy from another open source project must be compatible with LGPL.

16. Finally: That nasty word **Documentation**. While nobody expects a developer to spend his time writing the documentation for the manual, some documentation is necessary. Writing a simple proposal in a Wiki page on dev.tiki.org stating what you want to accomplish and how you plan to do it does several things. First, it will help you to clarify your thoughts. Second, it should provide you and others with some specific goals. The most important thing is that it will provide others with a place where they can comment on your ideas. Documenting the source code is also **important!** Comments allow the unskilled novice to become better.

Other development references:

- DevTips
- Commit Code
- SVNTips
- Introduction to Tiki Code Layout

**Aliases**

:Get Started | GetStarted