

Разработка домашних страниц

Основы HTML и CSS для неспециалистов

Алексей Куприянов

<http://dhhse.narod.ru/faculty/q/>

tinea@eu.spb.ru

В чем я вижу цель занятия?

- Заразить всех идеей разработки ресурсных страниц для читаемых на кафедре курсов и показать, как создавать такие страницы.

О чем в связи с этим предстоит рассказать?

о том, что такое гипертекстовый документ
как сделать так, чтобы документ был понятен
компьютеру
как сделать так, чтобы документ радовал глаз
и чтобы им было удобно пользоваться

О чем мы будем сегодня говорить?

1. О выделении в документе содержания (“контента”), логической структуры и графического представления.
2. О том, какого рода контент нужен нам как преподавателям
3. О том, как внести в документ разметку, отражающую логическую структуру
4. О том, как управлять графическим представлением документа, т. е.
 - О типографике
 - О цвете
 - Об использовании графики
5. Об информационной структуре сайта
6. Об основах web-usability

Как сделать документ понятным компьютеру?

- Надо правильно назвать его
- Надо внести в текст документа разметку, отражающую его логическую структуру

Основные этапы разработки домашней страницы

- Готовим контент.
- Разрабатываем информационную схему.
- Исходя из информационной схемы, определяем функциональные типы страниц (тип страницы отражает характер связей с другими страницами).
- Разрабатываем дизайн (для каждого типа страниц—свой).
 - Цветовое решение, гарнитуры / кегли текстовых элементов, графика.
- Пишем код.
 - Пишем HTML код страниц.
 - Пишем код таблицы стилей.
 - Готовим графику.
- Смотрим разными браузерами, проверяем, все ли ссылки работают.

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - Контейнеры
Имеется "открывающий" *tag* и "закрывающий" *tag*.
Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).
 - `<tagname>Содержимое контейнера</tagname>`
 - `<h1>Заголовок первого уровня</h1>`
 - `<p>Текст абзаца</p>`
 - Пустые
(закрывающий *tag* отсутствует)
 - `<tagname>`
 - `
` разрыв строки

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - **Контейнеры**
Имеется “открывающий” *tag* и “закрывающий” *tag*. Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).
 - **<tagname>** Содержимое контейнера **</tagname>**
 - **<h1>** Заголовок первого уровня **</h1>**
 - **<p>** Текст абзаца **</p>**
 - Пустые
(закрывающий *tag* отсутствует)
 - **<tagname>**
 - **
** разрыв строки

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - **Контейнеры**
Имеется "**открывающий**" *tag* и "**закрывающий**" *tag*.
Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).
 - **<tagname>** Содержимое контейнера **</tagname>**
 - **<h1>** Заголовок первого уровня **</h1>**
 - **<p>** Текст абзаца **</p>**
 - Пустые
(закрывающий *tag* отсутствует)
 - **<tagname>**
 - **
** разрыв строки

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - **Контейнеры**

Имеется "открывающий" *tag* и "закрывающий" *tag*. Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).

 - `<tagname>` Содержимое контейнера `</tagname>`
 - `<h1>` Заголовок первого уровня `</h1>`
 - `<p>` Текст абзаца `</p>`
 - Пустые
(закрывающий *tag* отсутствует)
 - `<tagname>`
 - `
` разрыв строки

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - **Контейнеры**
Имеется "открывающий" *tag* и "закрывающий" *tag*.
Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).
 - `<tagname>Содержимое контейнера</tagname>`
 - `<h1>Заголовок первого уровня</h1>`
 - `<p>Текст абзаца</p>`
 - Пустые
(закрывающий *tag* отсутствует)
 - `<tagname>`
 - `
` разрыв строки

Как пишутся команды (*tags*) HTML?

- С точки зрения написания, команды HTML делятся на
 - Контейнеры
Имеется "открывающий" *tag* и "закрывающий" *tag*.
Закрывающий *tag* отличается от открывающего наличием *slash* (наклонной черты перед именем *tag*).
 - `<tagname>Содержимое контейнера</tagname>`
 - `<h1>Заголовок первого уровня</h1>`
 - `<p>Текст абзаца</p>`
 - Пустые
(закрывающий *tag* отсутствует)
 - `<tagname>`
 - `
` (разрыв строки)

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают имя команды, параметры и значения параметров
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и много
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают **ИМЯ КОМАНДЫ**, параметры и значения параметров
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и много
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают имя команды, **параметры** и значения параметров
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и много
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают имя команды, параметры и **значения параметров**
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и много
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают имя команды, параметры и значения параметров
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и **МНОГО**
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Как устроена отдельная команда (*tag*) HTML?

- В составе команды различают имя команды, параметры и значения параметров
 - `<tagname attribute="value">...</tagname>`
 - `...`
- Параметров может быть и **МНОГО**
 - `<tagname attribute1="value" ... attributek="value">...`
 - `...`

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.

Плохо: `<p>`Раз абзац `<p>`Два абзац

- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.

Хорошо: `<p>`Раз абзац`</p>``<p>`Два абзаца`</p>`

- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.

Хорошо: `<p>`Раз абзац`</p>``<p>`Два абзаца`</p>`

- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Плохо: `<p> Текст Ссылка</p>`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Плохо: `<p> Текст`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Плохо: `<p> Текст Ссылка</p>`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Хорошо: `<p> Текст Ссылка</p>`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Хорошо: `<p> Текст`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.

Хорошо: `<p> Текст Ссылка</p>`

- Значения параметров следует заключать в кавычки.

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающий *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Плохо: `Ссылка`

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающий *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Хорошо: `Ссылка`

Три золотых правила написания команд (*tags*) HTML

- Контейнеры всегда должны заканчиваться на закрывающей *tag*.
- Контейнеры могут быть правильным образом вложены друг в друга, но не должны перекрываться.
- Значения параметров следует заключать в кавычки.

Хорошо: `Ссылка`

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных двоеточием.
 - attribute: value
 - color: #000
- Когда параметров стиля много, они отделяются друг от друга точкой с запятой.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из **параметра стиля** и значения параметра, разделенных двоеточием.
 - **attribute**: value
 - **color**: #000
- Когда параметров стиля много, они отделяются друг от друга точкой с запятой.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных двоеточием.
 - attribute: value
 - color: #000
- Когда параметров стиля много, они отделяются друг от друга точкой с запятой.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных **двоеточием**.
 - attribute: value
 - color: #000
- Когда параметров стиля много, они отделяются друг от друга точкой с запятой.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных двоеточием.
 - attribute: value
 - color: #000
- Когда параметров стиля **много**, они отделяются друг от друга точкой с запятой.
 - **attribute₁: value; ...; attribute_k: value**
 - **color: #000; background-color: #ccc; font-size: 70%**

Как устроены команды CSS?

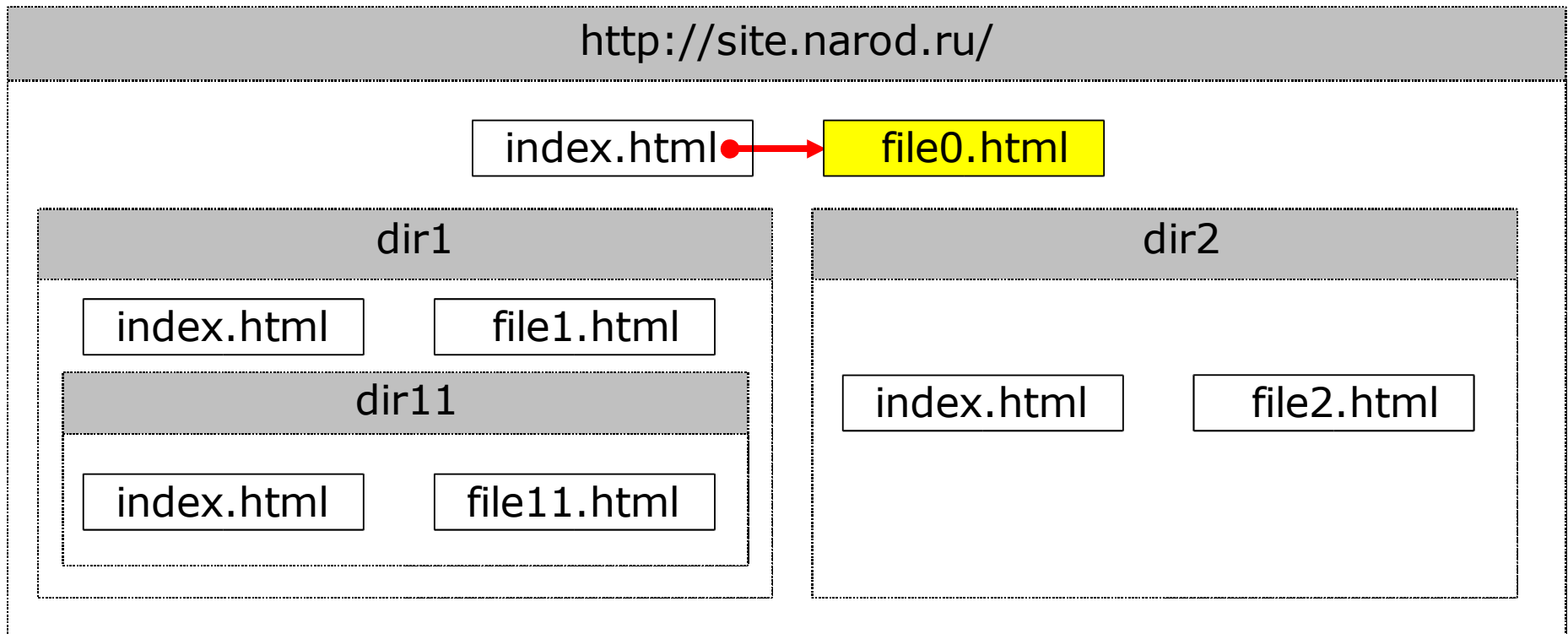
- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных двоеточием.
 - attribute: value
 - color: #000
- Когда параметров стиля **много**, они отделяются друг от друга точкой с запятой.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

Как устроены команды CSS?

- Команды CSS во многом похожи на команды HTML.
- Каждая команда состоит из параметра стиля и значения параметра, разделенных двоеточием.
 - attribute: value
 - color: #000
- Когда параметров стиля **много**, они отделяются друг от друга **точкой с запятой**.
 - attribute₁: value; ...; attribute_k: value
 - color: #000; background-color: #ccc; font-size: 70%

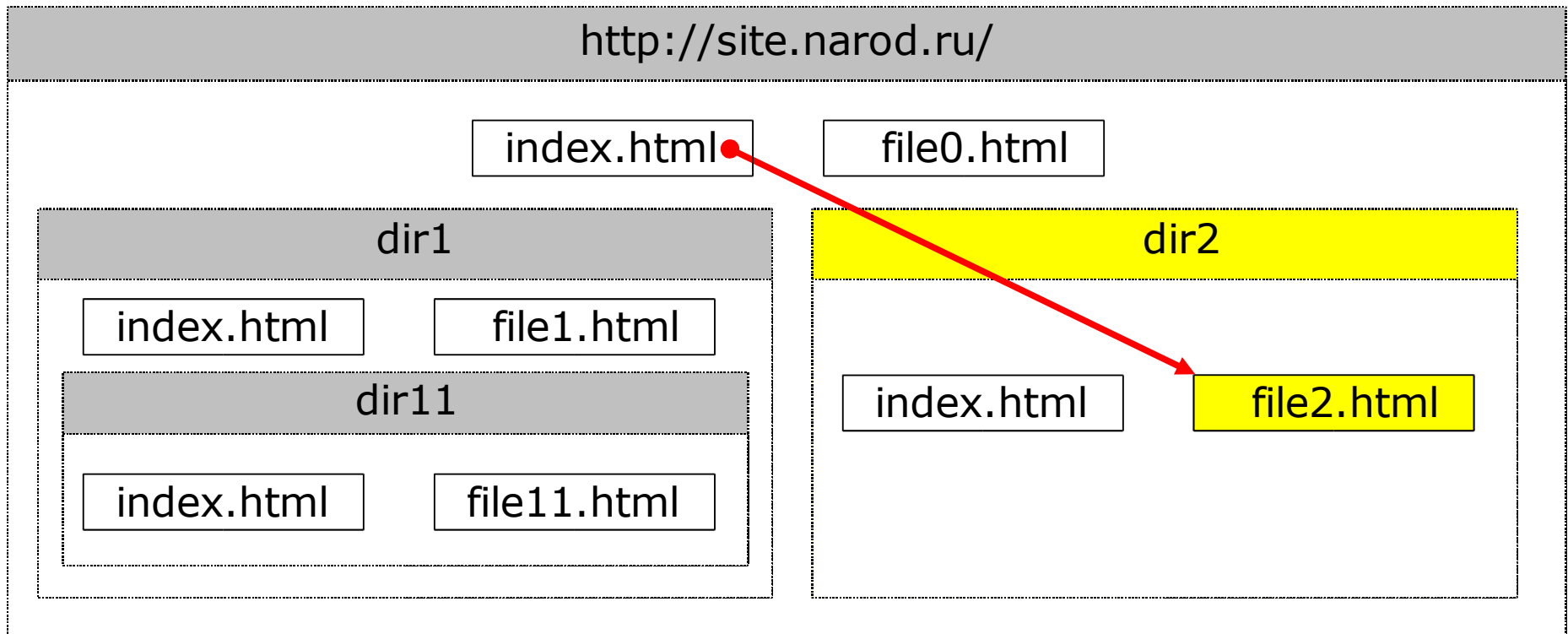
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/file0.html>
 - Относительно корневого каталога </file0.html>
 - Относительные <file0.html>



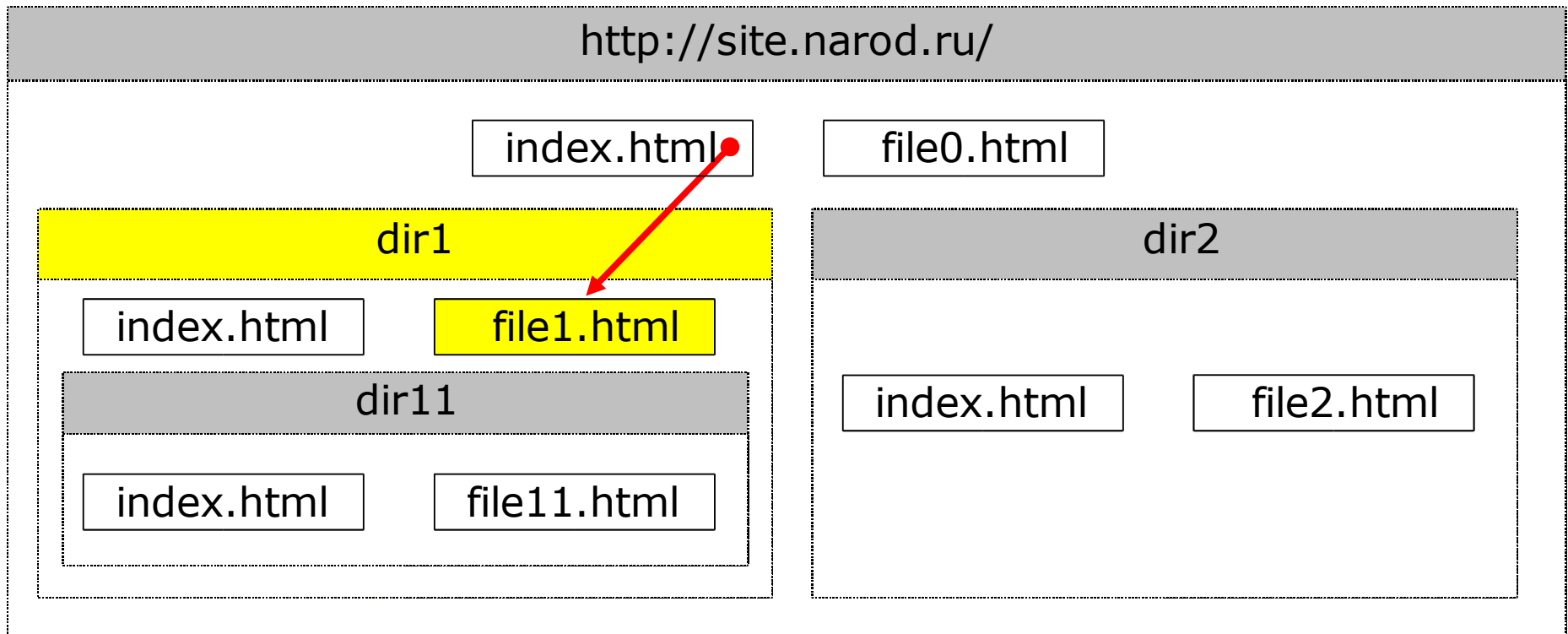
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir2/file2.html>
 - Относительно корневого каталога </dir2/file2.html>
 - Относительные <dir2/file2.html>



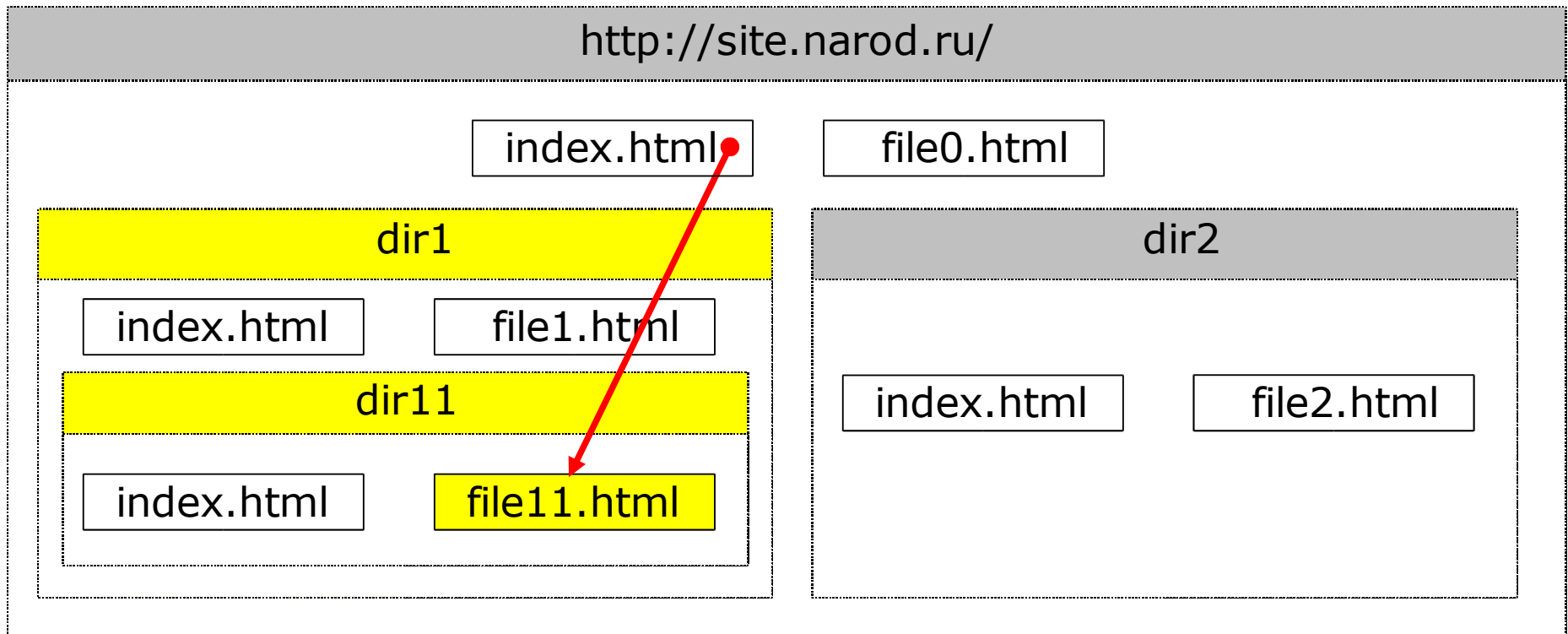
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/file1.html>
 - Относительно корневого каталога </dir1/file1.html>
 - Относительные <dir1/file1.html>



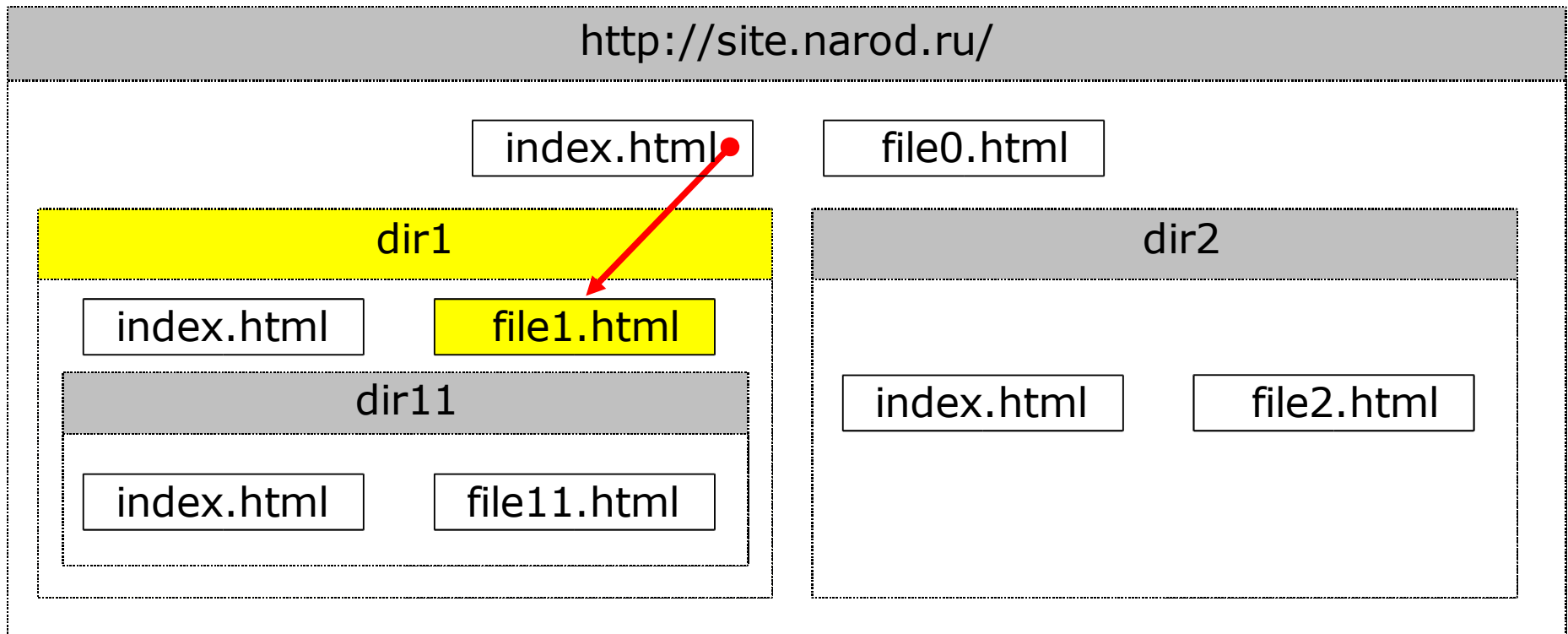
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/dir11/file11.html>
 - Относительно корневого каталога </dir1/dir11/file11.html>
 - Относительные <dir1/dir11/file2.html>



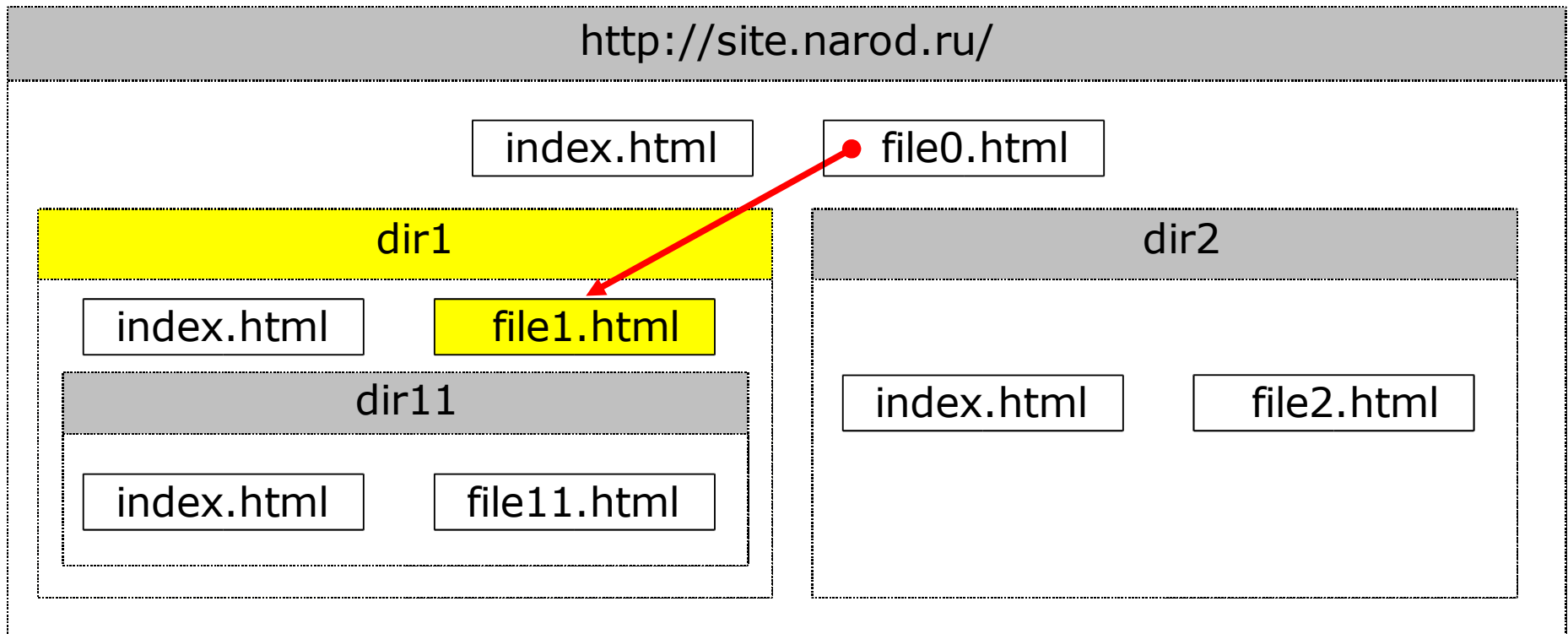
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/file1.html>
 - Относительно корневого каталога </dir1/file1.html>
 - Относительные <dir1/file1.html>



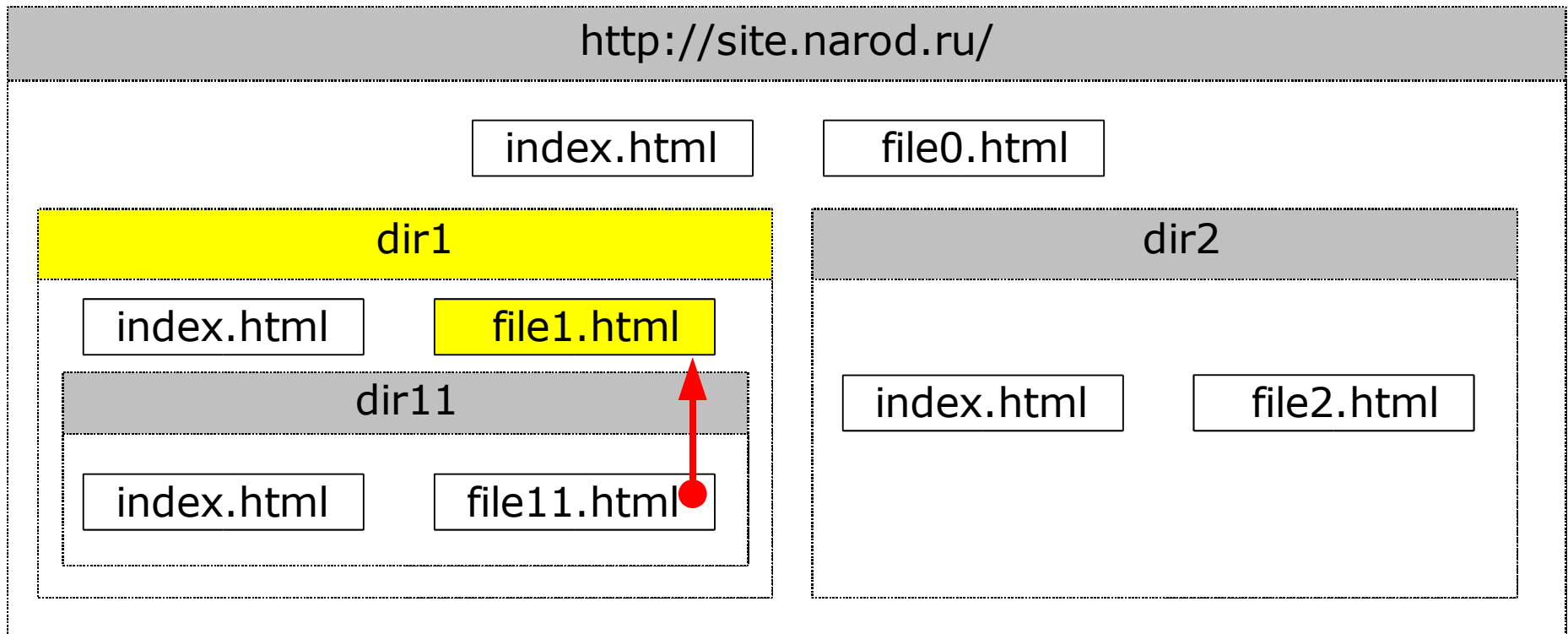
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/file1.html>
 - Относительно корневого каталога </dir1/file1.html>
 - Относительные <dir1/file1.html>



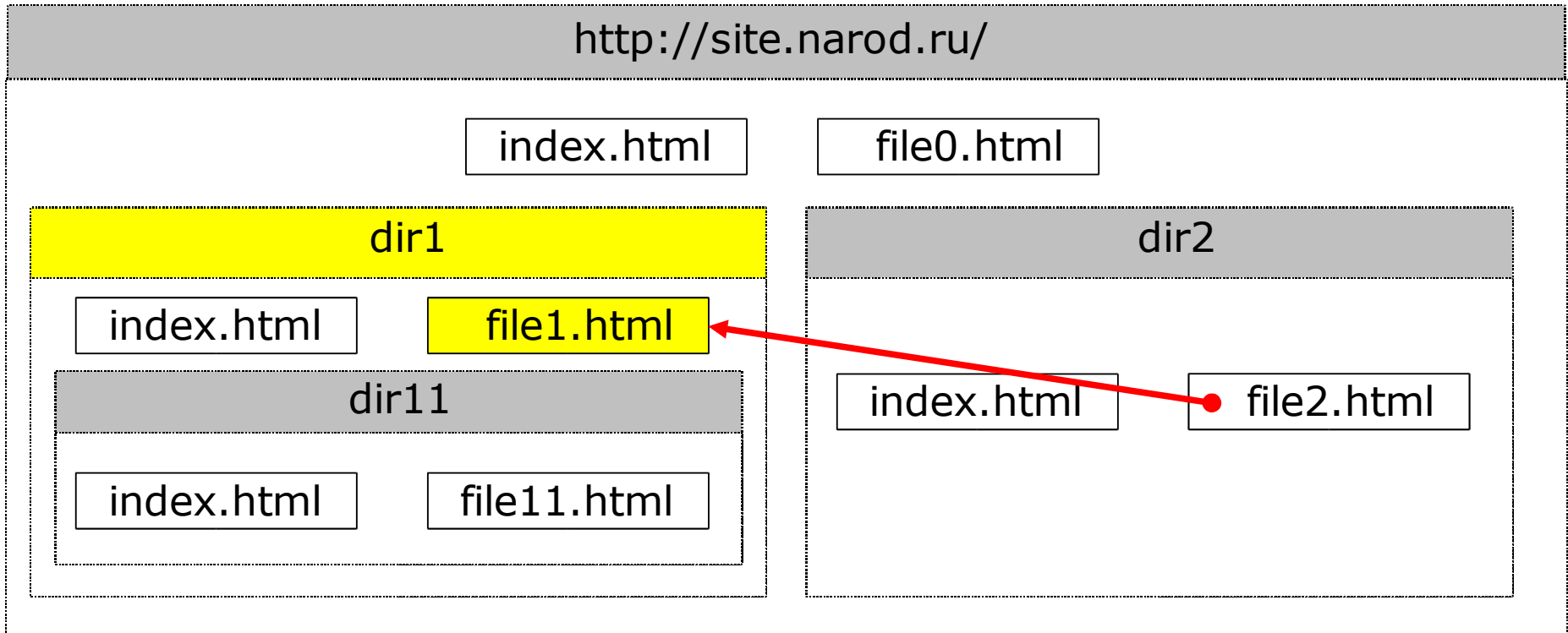
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/file1.html>
 - Относительно корневого каталога </dir1/file1.html>
 - Относительные <../file1.html>



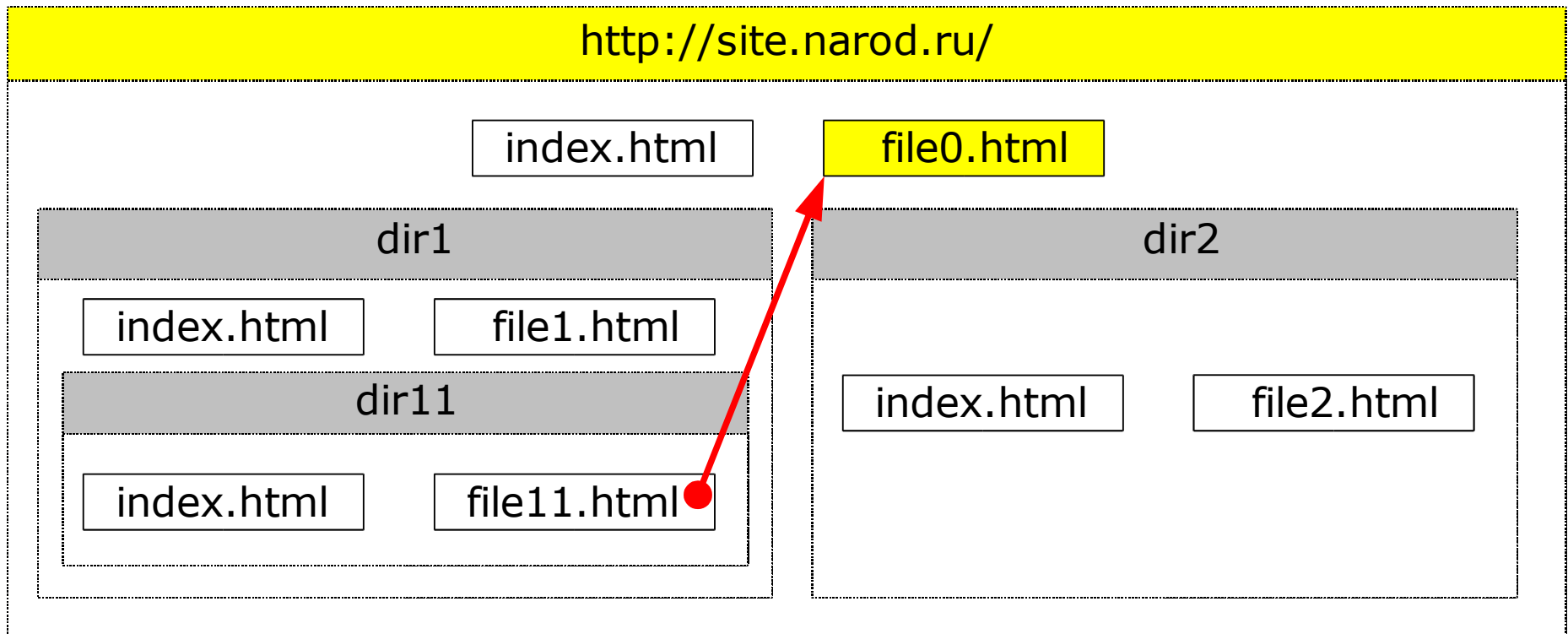
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir1/file1.html>
 - Относительно корневого каталога </dir1/file1.html>
 - Относительные <../dir1/file1.html>



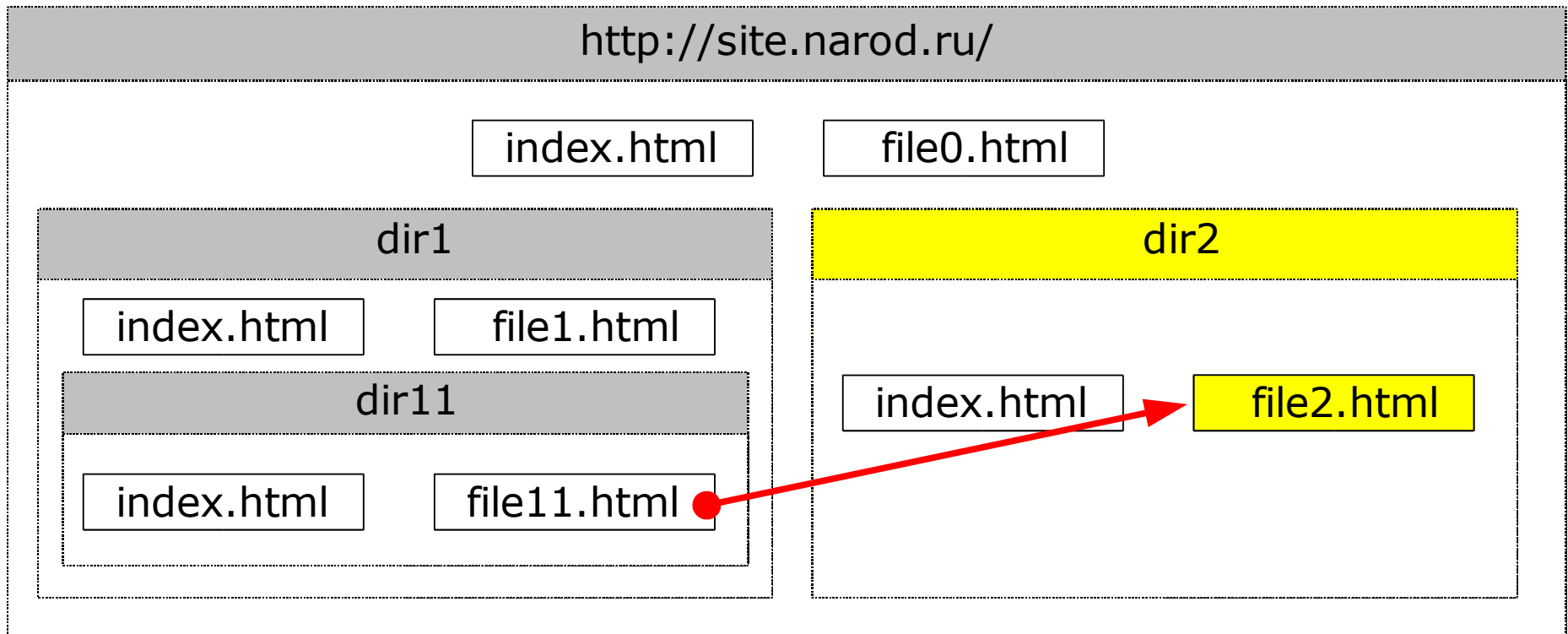
Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/file0.html>
 - Относительно корневого каталога </file0.html>
 - Относительные <../../file0.html>



Какими бывают адреса в гиперссылках?

- Адреса в гиперссылках:
 - Абсолютные <http://site.narod.ru/dir2/file2.html>
 - Относительно корневого каталога </dir2/file2.html>
 - Относительные <../../dir2/file1.html>



Чего избегаем, к чему стремимся?

- Избегаем:

- Создавать и править файлы прямо на сервере, используя онлайн-интерфейсы
- Обилия картинок
- Использования таблиц для позиционирования текста
- Нестандартных цветов для ссылок
- Принудительного открытия ссылок в отдельном окне

- Стремимся:

- Создавать и править файлы на локальной машине и загружать их на сервер в уже готовом виде
- В каждую директорию поместить файл по имени *index.html* или *index.htm*
- У каждой картинке заполнять параметры *alt* и *title*