**Вопросы 💎**

1. Как сделать так, чтобы при просмотре на телефоне текст стал читаемым, а картинка - большой?

To prevent Android and iOS mobile browsers from automatically resizing website pages, a special <meta> tag with the name="viewport" attribute is used. In this tag it is allowed to set a specific value for the width and initial-scale parameters:

**<meta name="viewport" content="width=device-width, initial-scale=1">**

Here **initial-scale=1** says that the page size in the browser will be equal to 100% of the viewport size. This means that 1 pixel of the viewport = 1 pixel of CSS.

**width=device-width** says that the page width will be equal to 100% of the width of any browser window. That is, the page width of the site matches the width of the device, so it doesn't need to be scaled.

In our example we can use e.g.:

<meta name="viewport" content="initial-scale=2.0, width=device-width">

Here the content in the browser window will be 2x the physical size.

1. В чём разница между отзывчивым и адаптивным веб-дизайном?

The main difference between these techniques is that we use one layout for all devices in case of responsive design, and one layout for each type of device in case of adaptive design.

A key feature of responsive web design is that due to the moving grid, the layout automatically reacts to changes in screen size, inflating or shrinking like a balloon. Whereas adaptive design designes a site with conditions that vary from device to device, based on several fixed-width layouts.

Put simply, responsive is fluid and adapts to the size of the screen no matter what the target device is. Responsive uses CSS media queries to change styles based on the target device such as display type, width, height, etc., and only one of these is necessary for the site to adapt to different screens.

Adaptive design, on the other hand, uses static layouts based on breakpoints that don’t respond once they’re initially loaded. Adaptive works to detect the screen size and load the appropriate layout for it – generally you would design an adaptive site for six common screen widths:

* 320
* 480
* 760
* 960
* 1200
* 1600

1. Какие величины лучше использовать для шрифтов в гибком дизайне?

**Mobile versions**

The font size of the main text in a mobile version should be at least 16px. The aim: to make the text on your phone (at a standard eye distance) as easy to read as on a page of a well-printed book (at a standard - usually slightly longer - eye distance). If your text inputs have a smaller font size than that, iOS browsers will zoom in on the left side of the text input, often obscuring the right side and forcing the user to manually zoom out after using the text box.

Understanding that different fonts can be more or less legible even at the exact same size, 16px is a good place to start when choosing your default mobile font size.

For secondary text, captions for margins and pictures, use a size a couple of values smaller: 13px or 14px, for example. It is not recommended to reduce the size by just one unit - then it will be easy to get confused with plain text. Less important text should be designed accordingly to deliberately emphasise its lesser importance.

Check how your design looks on the device itself.

Be familiar with Material Design and iOS standards.

**Desktop versions**

When picking a base size for a desktop website or web app, you can break down most designs into one of two types:

-Text-heavy pages. Articles, blogs, news, etc. These are pages where the primary purpose the user has on the page is to read. There is very little in terms of interaction – perhaps just clicking a few links.

-Interaction-heavy pages. Apps that involve more hovering, clicking, searching for an item in a list or table, editing, typing, etc. There may be plenty of text on the page, but it’s not stuff you read straight through like a book.

Long story short, for text-heavy pages, you want *larger font sizes*. If folks are reading for long periods of time, *be nice*: don’t make them strain their eyes. Now, each font is different, even at the same size, but we’re talking:

* **16px** – absolute minimum for text-heavy pages
* **18px** – a better font size to start with. You’re not printing out a single-spaced Word document; you’re writing for people sitting a couple feet from their decade-old monitors.
* **20px+** – may feel awkwardly large at first, but is *always* worth trying out in your design app.

Now, for interaction-heavy pages, smaller text sizes are perfectly acceptable. In fact, depending on the amount of data your user is taking in at once, even 18px text is uncomfortably large.

1. Какой вид верстки использован на этой картинке? К какой категории шаблонов он относится?

Responsive design and a Layout Shifter template are used here.

1. Как задать стили для экранов шириной от 800 до 1200 пикселей?

@media (min-width: 800px) and (max-width: 1200px) {/\* CSS-стили \*/;}

1. Приведите минимум 2 примера как подключать медиазапросы?

In general a media query consists of a keyword describing the device type (optional parameter) and an expression checking the characteristics of the device. The device width is the most commonly checked characteristic. Mediaquery is a logical expression that returns true or false. A media query can be done in various ways:

* Using HTML

<link rel="stylesheet" media="screen and (color)" href="example.css">

* Using the @import rule inside the <style> element or an external stylesheet:

@import url(color.css) screen and (color);

* Inside the webpage code:

<style> @media (max-width: 600px) { #sidebar {display: none;} } </style>

* Inside the CSS stylesheet:

@media (max-width: 600px) { #sidebar {display: none;} }

1. Как можно задавать гибкие изображения?

One solution is to use max-width in CSS:

img {max-width: 100%;}

If the max-width property is set to 100%, the image will be resized if necessary, but will never be enlarged to a size greater than its original size.

The maximum width of an image is 100% of the screen or browser window width, so the smaller the width, the smaller the image. Note that max-width is not supported in IE, so use width: 100% there.

The <picture> HTML element allows you to define different images for different browser window sizes.

1. Как задать стили только для landscape поворота экрана? И что вообще такое landscape и чем он отличается от portrait?

We can use the orientation function that checks whether the page is displayed in portrait or landscape mode.

@media (orientation: landscape) { #background-image { background: url(image1.png) no-repeat; }

Landscape is a mode where the opposite of viewport width is greater than its height.

Portrait is on the contrary a mode where the height of the viewport is greater than the width.

1. Назовите минимум 3 способа как можно тестировать, как выглядит сайт при разных размерах экранов?

One of the ways to view the site in different screen resolutions is through the built-in features of browsers. DevTools' Device Mode offers an easy way for developers to simulate mobile devices within the Chrome browser. Right-click on the desired site and select View Code. An additional panel with the site codes appears, where you click on the special tab to enable viewing of the site in different resolutions. To select different mobile devices, use the drop-down menu. The right-hand side also indicates the screen resolution of the selected device. To unfold the screen of the device - use the corresponding button.

Another way is to view the screen resolution of a website in Screenfly. This is one of the most convenient (and free) services at the moment, with which you can quickly view the site not only on the basic, but generally on different screen resolutions. You need to enter the website to be checked into the suggested field and get the result.

The simplest way but not the recommended one ist o look for the basic screen resolutions for the site through simple Windows tools, by changing the screen resolution under Screen Settings.

The beauty of the tool Responsinator lies in its simplicity. Just type in your web page's URL and this free, browser-based tool shows you how your web page renders in the most popular screen shapes and sizes.

1. Самостоятельно изучите, как можно подключить несколько картинок разных размеров через один тег <img>?

There are two new attributes - srcset and sizes - to add additional tagged images for the browser to choose from.

<img srcset="elva-fairy-320w.jpg 320w,

elva-fairy-480w.jpg 480w,

elva-fairy-800w.jpg 800w"

sizes="(max-width: 320px) 280px,

(max-width: 480px) 440px,

800px"

src="elva-fairy-800w.jpg" alt="Elva dressed as a fairy">

srcset includes the names of the images, among which the browser will select the right one and their sizes. In front of each comma are the parts of the values in this order:

* Image name (elva-fairy-480w.jpg.)
* Space.
* The actual width of the image in pixels (480w) - note that w is used here instead of px. This is the actual width of the image which can be viewed in the image properties on your computer

Sizes defines a list of media expressions (e.g. screen width) and specifies the preferred image width when a certain media expression is true - this is what we talked about above. In our case, we write before each comma:

* Media condition ((max-width:480px)) - You can read more about this in the CSS topic, but for now let's say that the media condition describes a possible screen state. In this case, we're saying "when viewport width is less than or equal to 480 pixels".
* Space.
* The slot width (in the original "width of the slot") occupied by the image when the media condition is true. (440px)