レスポンシブWebデザイン（RWD）について　no.03　　　　　　2017.12.28

(1) トップページのレスポンシブ対応

　　　<http://www.shoai.ne.jp/pre/pre-hirakata-s/index.html>

スマホ画面は、メニューボタンを追加してon・offでサイドメニューを表示できるようにして、コンテンツ重視の画面とする。（下の画像）

パソコン画面は従来のままとする。

技術的には

①メディアクエリーは画面幅768pxをブレークポイントする

②トップページのベースとなる外部cssのsouthtop.css(通常のhira-mにあたるもの)

 をパソコン用はsouthtop-dt.css、スマホ用はsouthtop-sp.cssに分ける

<link href="southtop-dt.css" rel="stylesheet" media="print,screen and (min-width: 768px)">

<link href="southtop-sp.css" rel="stylesheet" media="only screen and (max-width: 767px)">

③viewportはユーザーエージェント(UA)情報を基にjQueryで振り分ける

<script src="jquery/viewport.js"></script>

【jQueryの内容】

/\* スマホとタブレットの切り分け \*/

$(function(){

 var ua = navigator.userAgent;

 if((ua.indexOf('iPhone') > 0) || (ua.indexOf('Android') > 0 && ua.indexOf('Mobile') > 0) || ユーザー

 $('head').prepend('<meta name="viewport" content="width=device-width,initial-scale=1">');

 }

 else{

 $('head').prepend('<meta name="viewport" content="width=900">');

 }

});

④メニューボタンを追加してon・offでサイドメニューを表示できる

<script>

/\* dropdownコントロール \*/

$(function(){

// $("#menu").css("display","block");

 $("#menu-bt").on("click", function() {

 $("#menu").slideToggle(300);

 $(this).toggleClass("active");

 if($(this).hasClass("active")){

 $("#drower").attr('src','topimage/spmenu-close.png');

 }else{

 $("#drower").attr('src','topimage/spmenu-open-e.png');

 }

 });

});

</script>

右画面は最初に出るトップページのメニュー画面

左画面はMENUボタンを押した後のメニュー画面



　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　以上