**Source Code:**

package com.icinbank.onlinebanking;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class OnlineBankingApplication {

public static void main(String[] args) {

SpringApplication.run(OnlineBankingApplication.class, args);

}

}

package com.icinbank.onlinebanking;

import org.junit.Test;

import org.junit.runner.RunWith;

import org.springframework.boot.test.context.SpringBootTest;

import org.springframework.test.context.junit4.SpringRunner;

@RunWith(SpringRunner.class)

@SpringBootTest

public class OnlineBankingApplicationTests {

@Test

public void contextLoads() {

}

}

package com.icinbank.onlinebanking.Configuration;

import org.springframework.core.Ordered;

import org.springframework.core.annotation.Order;

import org.springframework.stereotype.Component;

import jakarta.servlet.Filter;

import jakarta.servlet.FilterChain;

import jakarta.servlet.FilterConfig;

import jakarta.servlet.ServletRequest;

import jakarta.servlet.ServletResponse;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

@Component

@Order(Ordered.HIGHEST\_PRECEDENCE)

public class RequestFilter implements Filter {

public void doFilter(ServletRequest req, ServletResponse res, FilterChain chain) {

HttpServletResponse response = (HttpServletResponse) res;

HttpServletRequest request = (HttpServletRequest) req;

response.setHeader("Access-Control-Allow-Origin", "[http://localhost:4200](http://localhost:4200/)");

response.setHeader("Access-Control-Allow-Methods", "POST, PUT, GET, OPTIONS, DELETE");

response.setHeader("Access-Control-Allow-Headers", "x-requested-with");

response.setHeader("Access-Control-Max-Age", "3600");

response.setHeader("Access-Control-Allow-Credentials", "true");

if (!(request.getMethod().equalsIgnoreCase("OPTIONS"))) {

try {

chain.doFilter(req, res);

} catch (Exception e) {

e.printStackTrace();

}

} else {

System.out.println("Pre-flight");

response.setHeader("Access-Control-Allow-Methods", "POST,GET,DELETE");

response.setHeader("Access-Control-Max-Age", "3600");

response.setHeader("Access-Control-Allow-Headers", "authorization, content-type," +

"access-control-request-headers,access-control-request-method,accept,origin,authorization,x-requested-with");

response.setStatus(HttpServletResponse.SC\_OK);

}

}

public void init(FilterConfig filterConfig) {

}

public void destroy() {

}

}

package com.icinbank.onlinebanking.Configuration;

import com.icinbank.onlinebanking.Service.UserServiceImpl.UserSecurityService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;

import org.springframework.security.config.annotation.method.configuration.EnableGlobalMethodSecurity;

import org.springframework.security.config.annotation.web.builders.HttpSecurity;

import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;

import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

import org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder;

import org.springframework.security.web.util.matcher.AntPathRequestMatcher;

import java.security.SecureRandom;

@Configuration

@EnableWebSecurity

@EnableGlobalMethodSecurity(prePostEnabled = true)

public class SecurityConfig extends WebSecurityConfigurerAdapter {

// @Autowired

// private Environment env;

private static final String SALT = "salt"; // Salt should be protected carefully

private static final String[] PUBLIC\_MATCHERS = {

"/webjars/\*\*",

"/css/\*\*",

"/js/\*\*",

"/images/\*\*",

"/",

"/about/\*\*",

"/contact/\*\*",

"/error/\*\*/\*",

"/console/\*\*",

"/signup"

};

@Autowired

private UserSecurityService userSecurityService;

@Bean

public BCryptPasswordEncoder passwordEncoder() {

return new BCryptPasswordEncoder(12, new SecureRandom(SALT.getBytes()));

}

@Override

protected void configure(HttpSecurity http) throws Exception {

http

.authorizeRequests().

// antMatchers("/\*\*").

antMatchers(PUBLIC\_MATCHERS).

permitAll().anyRequest().authenticated();

http

.csrf().disable().cors().disable()

.formLogin().failureUrl("/index?error").defaultSuccessUrl("/userFront").loginPage("/index").permitAll()

.and()

.logout().logoutRequestMatcher(new AntPathRequestMatcher("/logout")).logoutSuccessUrl("/index?logout").deleteCookies("remember-me").permitAll()

.and()

.rememberMe();

}

@Autowired

public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception {

// auth.inMemoryAuthentication().withUser("user").password("password").roles("USER"); //This is in-memory authentication

auth.userDetailsService(userSecurityService).passwordEncoder(passwordEncoder());

}

}

package com.icinbank.onlinebanking.Controller;

import com.icinbank.onlinebanking.Entity.Appointment;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Service.AppointmentService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import java.security.Principal;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Date;

@Controller

@RequestMapping("/appointment")

public class AppointmentController {

@Autowired

private AppointmentService appointmentService;

@Autowired

private UserService userService;

@RequestMapping(value = "/create", method = RequestMethod.GET)

public String createAppointment(Model model) {

Appointment appointment = new Appointment();

model.addAttribute("appointment", appointment);

model.addAttribute("dateString", "");

return "appointment";

}

@RequestMapping(value = "/create", method = RequestMethod.POST)

public String createAppointmentPost(@ModelAttribute("appointment") Appointment appointment, @ModelAttribute("dateString") String date, Model model, Principal principal) throws ParseException {

SimpleDateFormat format1 = new SimpleDateFormat("yyyy-MM-dd hh:mm");

Date d1 = format1.parse(date);

appointment.setDate(d1);

User user = userService.findByUsername(principal.getName());

appointment.setUser(user);

appointmentService.createAppointment(appointment);

return "redirect:/userFront";

}

}

package com.icinbank.onlinebanking.Controller;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Service.AccountService;

import com.icinbank.onlinebanking.Service.TransactionService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import java.security.Principal;

import java.util.List;

@Controller

@RequestMapping("/account")

public class AccountController {

@Autowired

private UserService userService;

@Autowired

private AccountService accountService;

@Autowired

private TransactionService transactionService;

@RequestMapping("/primaryAccount")

public String primaryAccount(Model model, Principal principal) {

List<PrimaryTransaction> primaryTransactionList = transactionService.findPrimaryTransactionList(principal.getName());

User user = userService.findByUsername(principal.getName());

PrimaryAccount primaryAccount = user.getPrimaryAccount();

model.addAttribute("primaryAccount", primaryAccount);

model.addAttribute("primaryTransactionList", primaryTransactionList);

return "primaryAccount";

}

@RequestMapping("/savingsAccount")

public String savingsAccount(Model model, Principal principal) {

List<SavingsTransaction> savingsTransactionList = transactionService.findSavingsTransactionList(principal.getName());

User user = userService.findByUsername(principal.getName());

SavingsAccount savingsAccount = user.getSavingsAccount();

model.addAttribute("savingsAccount", savingsAccount);

model.addAttribute("savingsTransactionList", savingsTransactionList);

return "savingsAccount";

}

@RequestMapping(value = "/deposit", method = RequestMethod.GET)

public String deposit(Model model) {

model.addAttribute("accountType", "");

model.addAttribute("amount", "");

return "deposit";

}

@RequestMapping(value = "/deposit", method = RequestMethod.POST)

public String depositPOST(@ModelAttribute("amount") String amount, @ModelAttribute("accountType") String accountType, Principal principal) {

accountService.deposit(accountType, Double.parseDouble(amount), principal);

return "redirect:/userFront";

}

@RequestMapping(value = "/withdraw", method = RequestMethod.GET)

public String withdraw(Model model) {

model.addAttribute("accountType", "");

model.addAttribute("amount", "");

return "withdraw";

}

@RequestMapping(value = "/withdraw", method = RequestMethod.POST)

public String withdrawPOST(@ModelAttribute("amount") String amount, @ModelAttribute("accountType") String accountType, Principal principal) {

accountService.withdraw(accountType, Double.parseDouble(amount), principal);

return "redirect:/userFront";

}

}

package com.icinbank.onlinebanking.Controller;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.RoleDao;

import com.icinbank.onlinebanking.Security.UserRole;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import java.security.Principal;

import java.util.HashSet;

import java.util.Set;

@Controller

public class HomeController {

@Autowired

private UserService userService;

@Autowired

private RoleDao roleDao;

@RequestMapping("/")

public String home() {

return "redirect:/index";

}

@RequestMapping("/index")

public String index() {

return "index";

}

@GetMapping("/signup")

public String signup(Model model) {

User user = new User();

model.addAttribute("user", user);

return "signup";

}

@PostMapping("/signup")

public String signupPost(@ModelAttribute("user") User user, Model model) {

if (userService.checkUserExists(user.getUsername(), user.getEmail())) {

if (userService.checkEmailExists(user.getEmail())) {

model.addAttribute("emailExists", true);

}

if (userService.checkUsernameExists(user.getUsername())) {

model.addAttribute("usernameExists", true);

}

return "signup";

} else {

Set<UserRole> userRoles = new HashSet<>();

userRoles.add(new UserRole(user, roleDao.findByName("ROLE\_USER")));

userService.createUser(user, userRoles);

return "redirect:/";

}

}

@GetMapping("/userFront")

public String userFront(Principal principal, Model model) {

User user = userService.findByUsername(principal.getName());

PrimaryAccount primaryAccount = user.getPrimaryAccount();

SavingsAccount savingsAccount = user.getSavingsAccount();

model.addAttribute("primaryAccount", primaryAccount);

model.addAttribute("savingsAccount", savingsAccount);

return "userFront";

}

}

package com.icinbank.onlinebanking.Controller;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.Recipient;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Service.TransactionService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.transaction.annotation.Transactional;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RequestParam;

import java.security.Principal;

import java.util.List;

@Controller

@RequestMapping("/transfer")

public class TransferController {

@Autowired

private TransactionService transactionService;

@Autowired

private UserService userService;

@RequestMapping(value = "/betweenAccounts", method = RequestMethod.GET)

public String betweenAccounts(Model model) {

model.addAttribute("transferFrom", "");

model.addAttribute("transferTo", "");

model.addAttribute("amount", "");

return "betweenAccounts";

}

@RequestMapping(value = "/betweenAccounts", method = RequestMethod.POST)

public String betweenAccountsPost(

@ModelAttribute("transferFrom") String transferFrom,

@ModelAttribute("transferTo") String transferTo,

@ModelAttribute("amount") String amount,

Principal principal

) throws Exception {

User user = userService.findByUsername(principal.getName());

PrimaryAccount primaryAccount = user.getPrimaryAccount();

SavingsAccount savingsAccount = user.getSavingsAccount();

transactionService.betweenAccountsTransfer(transferFrom, transferTo, amount, primaryAccount, savingsAccount);

return "redirect:/userFront";

}

@RequestMapping(value = "/recipient", method = RequestMethod.GET)

public String recipient(Model model, Principal principal) {

List<Recipient> recipientList = transactionService.findRecipientList(principal);

Recipient recipient = new Recipient();

model.addAttribute("recipientList", recipientList);

model.addAttribute("recipient", recipient);

return "recipient";

}

@RequestMapping(value = "/recipient/save", method = RequestMethod.POST)

public String recipientPost(@ModelAttribute("recipient") Recipient recipient, Principal principal) {

User user = userService.findByUsername(principal.getName());

recipient.setUser(user);

transactionService.saveRecipient(recipient);

return "redirect:/transfer/recipient";

}

@RequestMapping(value = "/recipient/edit", method = RequestMethod.GET)

public String recipientEdit(@RequestParam(value = "recipientName") String recipientName, Model model, Principal principal) {

Recipient recipient = transactionService.findRecipientByName(recipientName);

List<Recipient> recipientList = transactionService.findRecipientList(principal);

model.addAttribute("recipientList", recipientList);

model.addAttribute("recipient", recipient);

return "recipient";

}

@RequestMapping(value = "/recipient/delete", method = RequestMethod.GET)

@Transactional

public String recipientDelete(@RequestParam(value = "recipientName") String recipientName, Model model, Principal principal) {

transactionService.deleteRecipientByName(recipientName);

List<Recipient> recipientList = transactionService.findRecipientList(principal);

Recipient recipient = new Recipient();

model.addAttribute("recipient", recipient);

model.addAttribute("recipientList", recipientList);

return "recipient";

}

@RequestMapping(value = "/toSomeoneElse", method = RequestMethod.GET)

public String toSomeoneElse(Model model, Principal principal) {

List<Recipient> recipientList = transactionService.findRecipientList(principal);

model.addAttribute("recipientList", recipientList);

model.addAttribute("accountType", "");

return "toSomeoneElse";

}

@RequestMapping(value = "/toSomeoneElse", method = RequestMethod.POST)

public String toSomeoneElsePost(@ModelAttribute("recipientName") String recipientName, @ModelAttribute("accountType") String accountType, @ModelAttribute("amount") String amount, Principal principal) {

User user = userService.findByUsername(principal.getName());

Recipient recipient = transactionService.findRecipientByName(recipientName);

transactionService.toSomeoneElseTransfer(recipient, accountType, amount, user.getPrimaryAccount(), user.getSavingsAccount());

return "redirect:/userFront";

}

}

package com.icinbank.onlinebanking.Controller;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import java.security.Principal;

@Controller

@RequestMapping("/user")

public class UserController {

@Autowired

private UserService userService;

@GetMapping("/profile")

public String profile(Principal principal, Model model) {

User user = userService.findByUsername(principal.getName());

model.addAttribute("user", user);

return "profile";

}

@PostMapping("/profile")

public String profilePost(@ModelAttribute("user") User newUser, Model model) {

User user = userService.findByUsername(newUser.getUsername());

user.setUsername(newUser.getUsername());

user.setFirstName(newUser.getFirstName());

user.setLastName(newUser.getLastName());

user.setEmail(newUser.getEmail());

user.setPhone(newUser.getPhone());

model.addAttribute("user", user);

userService.saveUser(user);

return "profile";

}

}

package com.icinbank.onlinebanking.Entity;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import java.util.Date;

@Entity

public class Appointment {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private Date date;

private String location;

private String description;

private boolean confirmed;

@ManyToOne

@JoinColumn(name = "user\_id")

private User user;

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public Date getDate() {

return date;

}

public void setDate(Date date) {

this.date = date;

}

public String getLocation() {

return location;

}

public void setLocation(String location) {

this.location = location;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public boolean isConfirmed() {

return confirmed;

}

public void setConfirmed(boolean confirmed) {

this.confirmed = confirmed;

}

@Override

public String toString() {

return "Appointment{" +

"id=" + id +

", date=" + date +

", location='" + location + '\'' +

", description='" + description + '\'' +

", user=" + user +

'}';

}

}

package com.icinbank.onlinebanking.Entity;

import com.fasterxml.jackson.annotation.JsonIgnore;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Entity;

import jakarta.persistence.FetchType;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import java.math.BigDecimal;

import java.util.List;

@Entity

public class PrimaryAccount {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private int accountNumber;

private BigDecimal accountBalance;

@OneToMany(mappedBy = "primaryAccount", cascade = CascadeType.ALL, fetch = FetchType.LAZY)

@JsonIgnore

private List<PrimaryTransaction> primaryTransactionList;

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public int getAccountNumber() {

return accountNumber;

}

public void setAccountNumber(int accountNumber) {

this.accountNumber = accountNumber;

}

public BigDecimal getAccountBalance() {

return accountBalance;

}

public void setAccountBalance(BigDecimal accountBalance) {

this.accountBalance = accountBalance;

}

public List<PrimaryTransaction> getPrimaryTransactionList() {

return primaryTransactionList;

}

public void setPrimaryTransactionList(List<PrimaryTransaction> primaryTransactionList) {

this.primaryTransactionList = primaryTransactionList;

}

}

package com.icinbank.onlinebanking.Entity;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import java.math.BigDecimal;

import java.util.Date;

@Entity

public class PrimaryTransaction {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private Date date;

private String description;

private String type;

private String status;

private double amount;

private BigDecimal availableBalance;

@ManyToOne

@JoinColumn(name = "primary\_account\_id")

private PrimaryAccount primaryAccount;

public PrimaryTransaction() {

}

public PrimaryTransaction(Date date, String description, String type, String status, double amount, BigDecimal availableBalance, PrimaryAccount primaryAccount) {

this.date = date;

this.description = description;

this.type = type;

this.status = status;

this.amount = amount;

this.availableBalance = availableBalance;

this.primaryAccount = primaryAccount;

}

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public Date getDate() {

return date;

}

public void setDate(Date date) {

this.date = date;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public double getAmount() {

return amount;

}

public void setAmount(double amount) {

this.amount = amount;

}

public BigDecimal getAvailableBalance() {

return availableBalance;

}

public void setAvailableBalance(BigDecimal availableBalance) {

this.availableBalance = availableBalance;

}

public PrimaryAccount getPrimaryAccount() {

return primaryAccount;

}

public void setPrimaryAccount(PrimaryAccount primaryAccount) {

this.primaryAccount = primaryAccount;

}

}

package com.icinbank.onlinebanking.Entity;

import com.fasterxml.jackson.annotation.JsonIgnore;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

@Entity

public class Recipient {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name;

private String email;

private String phone;

private String accountNumber;

private String description;

@ManyToOne

@JoinColumn(name = "user\_id")

@JsonIgnore

private User user;

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPhone() {

return phone;

}

public void setPhone(String phone) {

this.phone = phone;

}

public String getAccountNumber() {

return accountNumber;

}

public void setAccountNumber(String accountNumber) {

this.accountNumber = accountNumber;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

}

package com.icinbank.onlinebanking.Entity;

import com.fasterxml.jackson.annotation.JsonIgnore;

import com.icinbank.onlinebanking.Security.Authority;

import com.icinbank.onlinebanking.Security.UserRole;

import org.springframework.security.core.GrantedAuthority;

import org.springframework.security.core.userdetails.UserDetails;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.FetchType;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import jakarta.persistence.OneToOne;

import java.util.Collection;

import java.util.HashSet;

import java.util.List;

import java.util.Set;

@Entity

public class User implements UserDetails {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

@Column(name = "userId", nullable = false, updatable = false)

private Long userId;

private String username;

private String password;

private String firstName;

private String lastName;

@Column(name = "email", nullable = false, unique = true)

private String email;

private String phone;

private boolean enabled = true;

@OneToOne

private PrimaryAccount primaryAccount;

@OneToOne

private SavingsAccount savingsAccount;

@OneToMany(mappedBy = "user", cascade = CascadeType.ALL, fetch = FetchType.LAZY)

@JsonIgnore

private List<Appointment> appointmentList;

@OneToMany(mappedBy = "user", cascade = CascadeType.ALL, fetch = FetchType.LAZY)

private List<Recipient> recipientList;

@OneToMany(mappedBy = "user", cascade = CascadeType.ALL, fetch = FetchType.EAGER)

@JsonIgnore

private Set<UserRole> userRoles = new HashSet<>();

public Set<UserRole> getUserRoles() {

return userRoles;

}

public void setUserRoles(Set<UserRole> userRoles) {

this.userRoles = userRoles;

}

public Long getUserId() {

return userId;

}

public void setUserId(Long userId) {

this.userId = userId;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPhone() {

return phone;

}

public void setPhone(String phone) {

this.phone = phone;

}

public List<Appointment> getAppointmentList() {

return appointmentList;

}

public void setAppointmentList(List<Appointment> appointmentList) {

this.appointmentList = appointmentList;

}

public List<Recipient> getRecipientList() {

return recipientList;

}

public void setRecipientList(List<Recipient> recipientList) {

this.recipientList = recipientList;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public PrimaryAccount getPrimaryAccount() {

return primaryAccount;

}

public void setPrimaryAccount(PrimaryAccount primaryAccount) {

this.primaryAccount = primaryAccount;

}

public SavingsAccount getSavingsAccount() {

return savingsAccount;

}

public void setSavingsAccount(SavingsAccount savingsAccount) {

this.savingsAccount = savingsAccount;

}

@Override

public String toString() {

return "User{" +

"userId=" + userId +

", username='" + username + '\'' +

", password='" + password + '\'' +

", firstName='" + firstName + '\'' +

", lastName='" + lastName + '\'' +

", email='" + email + '\'' +

", phone='" + phone + '\'' +

", appointmentList=" + appointmentList +

", recipientList=" + recipientList +

", userRoles=" + userRoles +

'}';

}

@Override

public Collection<? extends GrantedAuthority> getAuthorities() {

Set<GrantedAuthority> authorities = new HashSet<>();

userRoles.forEach(ur -> authorities.add(new Authority(ur.getRole().getName())));

return authorities;

}

@Override

public boolean isAccountNonExpired() {

// TODO Auto-generated method stub

return true;

}

@Override

public boolean isAccountNonLocked() {

// TODO Auto-generated method stub

return true;

}

@Override

public boolean isCredentialsNonExpired() {

// TODO Auto-generated method stub

return true;

}

@Override

public boolean isEnabled() {

return enabled;

}

public void setEnabled(boolean enabled) {

this.enabled = enabled;

}

}

package com.icinbank.onlinebanking.Entity;

import com.fasterxml.jackson.annotation.JsonIgnore;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Entity;

import jakarta.persistence.FetchType;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import java.math.BigDecimal;

import java.util.List;

@Entity

public class SavingsAccount {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private int accountNumber;

private BigDecimal accountBalance;

@OneToMany(mappedBy = "savingsAccount", cascade = CascadeType.ALL, fetch = FetchType.LAZY)

@JsonIgnore

private List<SavingsTransaction> savingsTransactionList;

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public int getAccountNumber() {

return accountNumber;

}

public void setAccountNumber(int accountNumber) {

this.accountNumber = accountNumber;

}

public BigDecimal getAccountBalance() {

return accountBalance;

}

public void setAccountBalance(BigDecimal accountBalance) {

this.accountBalance = accountBalance;

}

public List<SavingsTransaction> getSavingsTransactionList() {

return savingsTransactionList;

}

public void setSavingsTransactionList(List<SavingsTransaction> savingsTransactionList) {

this.savingsTransactionList = savingsTransactionList;

}

package com.icinbank.onlinebanking.Entity;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import java.math.BigDecimal;

import java.util.Date;

@Entity

public class SavingsTransaction {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private Date date;

private String description;

private String type;

private String status;

private double amount;

private BigDecimal availableBalance;

@ManyToOne

@JoinColumn(name = "savings\_account\_id")

private SavingsAccount savingsAccount;

public SavingsTransaction() {

}

public SavingsTransaction(Date date, String description, String type, String status, double amount, BigDecimal availableBalance, SavingsAccount savingsAccount) {

this.date = date;

this.description = description;

this.type = type;

this.status = status;

this.amount = amount;

this.availableBalance = availableBalance;

this.savingsAccount = savingsAccount;

}

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public Date getDate() {

return date;

}

public void setDate(Date date) {

this.date = date;

}

public String getDescription() {

return description;

}

public void setDescription(String description) {

this.description = description;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public double getAmount() {

return amount;

}

public void setAmount(double amount) {

this.amount = amount;

}

public BigDecimal getAvailableBalance() {

return availableBalance;

}

public void setAvailableBalance(BigDecimal availableBalance) {

this.availableBalance = availableBalance;

}

public SavingsAccount getSavingsAccount() {

return savingsAccount;

}

public void setSavingsAccount(SavingsAccount savingsAccount) {

this.savingsAccount = savingsAccount;

}

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.Appointment;

import org.springframework.data.repository.CrudRepository;

import java.util.List;

public interface AppointmentDao extends CrudRepository<Appointment, Long> {

List<Appointment> findAll();

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import org.springframework.data.repository.CrudRepository;

public interface PrimaryAccountDao extends CrudRepository<PrimaryAccount, Long> {

PrimaryAccount findByAccountNumber(int accountNumber);

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import org.springframework.data.repository.CrudRepository;

import java.util.List;

public interface PrimaryTransactionDao extends CrudRepository<PrimaryTransaction, Long> {

List<PrimaryTransaction> findAll();

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.Recipient;

import org.springframework.data.repository.CrudRepository;

import java.util.List;

public interface RecipientDao extends CrudRepository<Recipient, Long> {

List<Recipient> findAll();

Recipient findByName(String recipientName);

void deleteByName(String recipientName);

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Security.Role;

import org.springframework.data.repository.CrudRepository;

public interface RoleDao extends CrudRepository<Role, Integer> {

Role findByName(String name);

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.User;

import org.springframework.data.repository.CrudRepository;

import java.util.List;

public interface UserDao extends CrudRepository<User, Long> {

User findByUsername(String username);

User findByEmail(String email);

List<User> findAll();

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import org.springframework.data.repository.CrudRepository;

public interface SavingsAccountDao extends CrudRepository<SavingsAccount, Long> {

SavingsAccount findByAccountNumber(int accountNumber);

}

package com.icinbank.onlinebanking.Repository;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import org.springframework.data.repository.CrudRepository;

import java.util.List;

public interface SavingsTransactionDao extends CrudRepository<SavingsTransaction, Long> {

List<SavingsTransaction> findAll();

}

package com.icinbank.onlinebanking.Resource;

import com.icinbank.onlinebanking.Entity.Appointment;

import com.icinbank.onlinebanking.Service.AppointmentService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.access.prepost.PreAuthorize;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import java.util.List;

@RestController

@RequestMapping("/api/appointment")

@PreAuthorize("hasRole('ADMIN')")

public class AppointmentResource {

@Autowired

private AppointmentService appointmentService;

@RequestMapping("/all")

public List<Appointment> findAppointmentList() {

List<Appointment> appointmentList = appointmentService.findAll();

return appointmentList;

}

@RequestMapping("/{id}/confirm")

public void confirmAppointment(@PathVariable("id") Long id) {

appointmentService.confirmAppointment(id);

}

}

package com.icinbank.onlinebanking.Resource;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Service.TransactionService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.access.prepost.PreAuthorize;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.RestController;

import java.util.List;

@RestController

@RequestMapping("/api")

@PreAuthorize("hasRole('ADMIN')")

public class UserResource {

@Autowired

private UserService userService;

@Autowired

private TransactionService transactionService;

@RequestMapping(value = "/user/all", method = RequestMethod.GET)

public List<User> userList() {

return userService.findUserList();

}

@RequestMapping(value = "/user/primary/transaction", method = RequestMethod.GET)

public List<PrimaryTransaction> getPrimaryTransactionList(@RequestParam("username") String username) {

return transactionService.findPrimaryTransactionList(username);

}

@RequestMapping(value = "/user/savings/transaction", method = RequestMethod.GET)

public List<SavingsTransaction> getSavingsTransactionList(@RequestParam("username") String username) {

return transactionService.findSavingsTransactionList(username);

}

@RequestMapping("/user/{username}/enable")

public void enableUser(@PathVariable("username") String username) {

userService.enableUser(username);

}

@RequestMapping("/user/{username}/disable")

public void diableUser(@PathVariable("username") String username) {

userService.disableUser(username);

}

}

package com.icinbank.onlinebanking.Security;

import org.springframework.security.core.GrantedAuthority;

public class Authority implements GrantedAuthority {

private final String authority;

public Authority(String authority) {

this.authority = authority;

}

@Override

public String getAuthority() {

return authority;

}

}

package com.icinbank.onlinebanking.Security;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Entity;

import jakarta.persistence.FetchType;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import java.util.HashSet;

import java.util.Set;

@Entity

public class Role {

@Id

private int roleId;

private String name;

@OneToMany(mappedBy = "role", cascade = CascadeType.ALL, fetch = FetchType.LAZY)

private Set<UserRole> userRoles = new HashSet<>();

public Role() {

}

public int getRoleId() {

return roleId;

}

public void setRoleId(int roleId) {

this.roleId = roleId;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public Set<UserRole> getUserRoles() {

return userRoles;

}

public void setUserRoles(Set<UserRole> userRoles) {

this.userRoles = userRoles;

}

}

package com.icinbank.onlinebanking.Security;

import com.icinbank.onlinebanking.Entity.User;

import jakarta.persistence.Entity;

import jakarta.persistence.FetchType;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import jakarta.persistence.Table;

@Entity

@Table(name = "user\_role")

public class UserRole {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private long userRoleId;

@ManyToOne(fetch = FetchType.EAGER)

@JoinColumn(name = "user\_id")

private User user;

@ManyToOne(fetch = FetchType.EAGER)

@JoinColumn(name = "role\_id")

private Role role;

public UserRole(User user, Role role) {

this.user = user;

this.role = role;

}

public UserRole() {

}

public long getUserRoleId() {

return userRoleId;

}

public void setUserRoleId(long userRoleId) {

this.userRoleId = userRoleId;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public Role getRole() {

return role;

}

public void setRole(Role role) {

this.role = role;

}

}

package com.icinbank.onlinebanking.Service;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import java.security.Principal;

public interface AccountService {

PrimaryAccount createPrimaryAccount();

SavingsAccount createSavingsAccount();

void deposit(String accountType, double amount, Principal principal);

void withdraw(String accountType, double amount, Principal principal);

}

package com.icinbank.onlinebanking.Service;

import com.icinbank.onlinebanking.Entity.Appointment;

import java.util.List;

public interface AppointmentService {

Appointment createAppointment(Appointment appointment);

List<Appointment> findAll();

Appointment findAppointment(Long id);

void confirmAppointment(Long id);

}

package com.icinbank.onlinebanking.Service;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import com.icinbank.onlinebanking.Entity.Recipient;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import java.security.Principal;

import java.util.List;

public interface TransactionService {

List<PrimaryTransaction> findPrimaryTransactionList(String username);

List<SavingsTransaction> findSavingsTransactionList(String username);

void savePrimaryDepositTransaction(PrimaryTransaction primaryTransaction);

void saveSavingsDepositTransaction(SavingsTransaction savingsTransaction);

void savePrimaryWithdrawTransaction(PrimaryTransaction primaryTransaction);

void saveSavingsWithdrawTransaction(SavingsTransaction savingsTransaction);

void betweenAccountsTransfer(String transferFrom, String transferTo, String amount, PrimaryAccount primaryAccount, SavingsAccount savingsAccount) throws Exception;

List<Recipient> findRecipientList(Principal principal);

Recipient saveRecipient(Recipient recipient);

Recipient findRecipientByName(String recipientName);

void deleteRecipientByName(String recipientName);

void toSomeoneElseTransfer(Recipient recipient, String accountType, String amount, PrimaryAccount primaryAccount, SavingsAccount savingsAccount);

}

package com.icinbank.onlinebanking.Service;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Security.UserRole;

import org.springframework.stereotype.Service;

import java.util.List;

import java.util.Set;

@Service("userDetailsService")

public interface UserService {

User findByUsername(String username);

User findByEmail(String email);

boolean checkUserExists(String username, String email);

boolean checkUsernameExists(String username);

boolean checkEmailExists(String email);

void save(User user);

User createUser(User user, Set<UserRole> userRoles);

User saveUser(User user);

List<User> findUserList();

void enableUser(String username);

void disableUser(String username);

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.PrimaryAccountDao;

import com.icinbank.onlinebanking.Repository.SavingsAccountDao;

import com.icinbank.onlinebanking.Service.AccountService;

import com.icinbank.onlinebanking.Service.TransactionService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import java.math.BigDecimal;

import java.security.Principal;

import java.util.Date;

@Service

public class AccountServiceImpl implements AccountService {

private static int nextAccountNumber = 11223101;

@Autowired

private PrimaryAccountDao primaryAccountDao;

@Autowired

private SavingsAccountDao savingsAccountDao;

@Autowired

private UserService userService;

@Autowired

private TransactionService transactionService;

public PrimaryAccount createPrimaryAccount() {

PrimaryAccount primaryAccount = new PrimaryAccount();

primaryAccount.setAccountBalance(new BigDecimal(0.0));

primaryAccount.setAccountNumber(accountGen());

primaryAccountDao.save(primaryAccount);

return primaryAccountDao.findByAccountNumber(primaryAccount.getAccountNumber());

}

public SavingsAccount createSavingsAccount() {

SavingsAccount savingsAccount = new SavingsAccount();

savingsAccount.setAccountBalance(new BigDecimal(0.0));

savingsAccount.setAccountNumber(accountGen());

savingsAccountDao.save(savingsAccount);

return savingsAccountDao.findByAccountNumber(savingsAccount.getAccountNumber());

}

public void deposit(String accountType, double amount, Principal principal) {

User user = userService.findByUsername(principal.getName());

if (accountType.equalsIgnoreCase("Primary")) {

PrimaryAccount primaryAccount = user.getPrimaryAccount();

primaryAccount.setAccountBalance(primaryAccount.getAccountBalance().add(new BigDecimal(amount)));

primaryAccountDao.save(primaryAccount);

Date date = new Date();

PrimaryTransaction primaryTransaction = new PrimaryTransaction(date, "Deposit to Primary Account", "Account", "Finished", amount, primaryAccount.getAccountBalance(), primaryAccount);

transactionService.savePrimaryDepositTransaction(primaryTransaction);

} else if (accountType.equalsIgnoreCase("Savings")) {

SavingsAccount savingsAccount = user.getSavingsAccount();

savingsAccount.setAccountBalance(savingsAccount.getAccountBalance().add(new BigDecimal(amount)));

savingsAccountDao.save(savingsAccount);

Date date = new Date();

SavingsTransaction savingsTransaction = new SavingsTransaction(date, "Deposit to savings Account", "Account", "Finished", amount, savingsAccount.getAccountBalance(), savingsAccount);

transactionService.saveSavingsDepositTransaction(savingsTransaction);

}

}

public void withdraw(String accountType, double amount, Principal principal) {

User user = userService.findByUsername(principal.getName());

if (accountType.equalsIgnoreCase("Primary")) {

PrimaryAccount primaryAccount = user.getPrimaryAccount();

primaryAccount.setAccountBalance(primaryAccount.getAccountBalance().subtract(new BigDecimal(amount)));

primaryAccountDao.save(primaryAccount);

Date date = new Date();

PrimaryTransaction primaryTransaction = new PrimaryTransaction(date, "Withdraw from Primary Account", "Account", "Finished", amount, primaryAccount.getAccountBalance(), primaryAccount);

transactionService.savePrimaryWithdrawTransaction(primaryTransaction);

} else if (accountType.equalsIgnoreCase("Savings")) {

SavingsAccount savingsAccount = user.getSavingsAccount();

savingsAccount.setAccountBalance(savingsAccount.getAccountBalance().subtract(new BigDecimal(amount)));

savingsAccountDao.save(savingsAccount);

Date date = new Date();

SavingsTransaction savingsTransaction = new SavingsTransaction(date, "Withdraw from savings Account", "Account", "Finished", amount, savingsAccount.getAccountBalance(), savingsAccount);

transactionService.saveSavingsWithdrawTransaction(savingsTransaction);

}

}

private int accountGen() {

return ++nextAccountNumber;

}

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.Appointment;

import com.icinbank.onlinebanking.Repository.AppointmentDao;

import com.icinbank.onlinebanking.Service.AppointmentService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class AppointmentServiceImpl implements AppointmentService {

@Autowired

private AppointmentDao appointmentDao;

public Appointment createAppointment(Appointment appointment) {

return appointmentDao.save(appointment);

}

public List<Appointment> findAll() {

return appointmentDao.findAll();

}

public Appointment findAppointment(Long id) {

return appointmentDao.findById(id).get();

}

public void confirmAppointment(Long id) {

Appointment appointment = findAppointment(id);

appointment.setConfirmed(true);

appointmentDao.save(appointment);

}

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.PrimaryAccount;

import com.icinbank.onlinebanking.Entity.PrimaryTransaction;

import com.icinbank.onlinebanking.Entity.Recipient;

import com.icinbank.onlinebanking.Entity.SavingsAccount;

import com.icinbank.onlinebanking.Entity.SavingsTransaction;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.PrimaryAccountDao;

import com.icinbank.onlinebanking.Repository.PrimaryTransactionDao;

import com.icinbank.onlinebanking.Repository.RecipientDao;

import com.icinbank.onlinebanking.Repository.SavingsAccountDao;

import com.icinbank.onlinebanking.Repository.SavingsTransactionDao;

import com.icinbank.onlinebanking.Service.TransactionService;

import com.icinbank.onlinebanking.Service.UserService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import java.math.BigDecimal;

import java.security.Principal;

import java.util.Date;

import java.util.List;

import java.util.stream.Collectors;

@Service

public class TransactionServiceImpl implements TransactionService {

@Autowired

private UserService userService;

@Autowired

private PrimaryTransactionDao primaryTransactionDao;

@Autowired

private SavingsTransactionDao savingsTransactionDao;

@Autowired

private PrimaryAccountDao primaryAccountDao;

@Autowired

private SavingsAccountDao savingsAccountDao;

@Autowired

private RecipientDao recipientDao;

public List<PrimaryTransaction> findPrimaryTransactionList(String username) {

User user = userService.findByUsername(username);

List<PrimaryTransaction> primaryTransactionList = user.getPrimaryAccount().getPrimaryTransactionList();

return primaryTransactionList;

}

public List<SavingsTransaction> findSavingsTransactionList(String username) {

User user = userService.findByUsername(username);

List<SavingsTransaction> savingsTransactionList = user.getSavingsAccount().getSavingsTransactionList();

return savingsTransactionList;

}

public void savePrimaryDepositTransaction(PrimaryTransaction primaryTransaction) {

primaryTransactionDao.save(primaryTransaction);

}

public void saveSavingsDepositTransaction(SavingsTransaction savingsTransaction) {

savingsTransactionDao.save(savingsTransaction);

}

public void savePrimaryWithdrawTransaction(PrimaryTransaction primaryTransaction) {

primaryTransactionDao.save(primaryTransaction);

}

public void saveSavingsWithdrawTransaction(SavingsTransaction savingsTransaction) {

savingsTransactionDao.save(savingsTransaction);

}

public void betweenAccountsTransfer(String transferFrom, String transferTo, String amount, PrimaryAccount primaryAccount, SavingsAccount savingsAccount) throws Exception {

if (transferFrom.equalsIgnoreCase("Primary") && transferTo.equalsIgnoreCase("Savings")) {

primaryAccount.setAccountBalance(primaryAccount.getAccountBalance().subtract(new BigDecimal(amount)));

savingsAccount.setAccountBalance(savingsAccount.getAccountBalance().add(new BigDecimal(amount)));

primaryAccountDao.save(primaryAccount);

savingsAccountDao.save(savingsAccount);

Date date = new Date();

PrimaryTransaction primaryTransaction = new PrimaryTransaction(date, "Between account transfer from " + transferFrom + " to " + transferTo, "Account", "Finished", Double.parseDouble(amount), primaryAccount.getAccountBalance(), primaryAccount);

primaryTransactionDao.save(primaryTransaction);

} else if (transferFrom.equalsIgnoreCase("Savings") && transferTo.equalsIgnoreCase("Primary")) {

primaryAccount.setAccountBalance(primaryAccount.getAccountBalance().add(new BigDecimal(amount)));

savingsAccount.setAccountBalance(savingsAccount.getAccountBalance().subtract(new BigDecimal(amount)));

primaryAccountDao.save(primaryAccount);

savingsAccountDao.save(savingsAccount);

Date date = new Date();

SavingsTransaction savingsTransaction = new SavingsTransaction(date, "Between account transfer from " + transferFrom + " to " + transferTo, "Transfer", "Finished", Double.parseDouble(amount), savingsAccount.getAccountBalance(), savingsAccount);

savingsTransactionDao.save(savingsTransaction);

} else {

throw new Exception("Invalid Transfer");

}

}

public List<Recipient> findRecipientList(Principal principal) {

String username = principal.getName();

List<Recipient> recipientList = recipientDao.findAll().stream()

.filter(recipient -> username.equals(recipient.getUser().getUsername()))

.collect(Collectors.toList());

return recipientList;

}

public Recipient saveRecipient(Recipient recipient) {

return recipientDao.save(recipient);

}

public Recipient findRecipientByName(String recipientName) {

return recipientDao.findByName(recipientName);

}

public void deleteRecipientByName(String recipientName) {

recipientDao.deleteByName(recipientName);

}

public void toSomeoneElseTransfer(Recipient recipient, String accountType, String amount, PrimaryAccount primaryAccount, SavingsAccount savingsAccount) {

if (accountType.equalsIgnoreCase("Primary")) {

primaryAccount.setAccountBalance(primaryAccount.getAccountBalance().subtract(new BigDecimal(amount)));

primaryAccountDao.save(primaryAccount);

Date date = new Date();

PrimaryTransaction primaryTransaction = new PrimaryTransaction(date, "Transfer to recipient " + recipient.getName(), "Transfer", "Finished", Double.parseDouble(amount), primaryAccount.getAccountBalance(), primaryAccount);

primaryTransactionDao.save(primaryTransaction);

} else if (accountType.equalsIgnoreCase("Savings")) {

savingsAccount.setAccountBalance(savingsAccount.getAccountBalance().subtract(new BigDecimal(amount)));

savingsAccountDao.save(savingsAccount);

Date date = new Date();

SavingsTransaction savingsTransaction = new SavingsTransaction(date, "Transfer to recipient " + recipient.getName(), "Transfer", "Finished", Double.parseDouble(amount), savingsAccount.getAccountBalance(), savingsAccount);

savingsTransactionDao.save(savingsTransaction);

}

}

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.UserDao;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.core.userdetails.UsernameNotFoundException;

import org.springframework.stereotype.Service;

@Service

public class UserSecurityService implements UserDetailsService {

private static final Logger LOG = LoggerFactory.getLogger(UserSecurityService.class);

@Autowired

private UserDao userDao;

@Override

public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {

User user = userDao.findByUsername(username);

if (null == user) {

LOG.warn("Username {} not found", username);

throw new UsernameNotFoundException("Username " + username + " not found");

}

return user;

}

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.UserDao;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.core.userdetails.UsernameNotFoundException;

import org.springframework.stereotype.Service;

@Service

public class UserSecurityServiceImpl implements UserDetailsService {

private static final Logger LOG = LoggerFactory.getLogger(UserSecurityServiceImpl.class);

@Autowired

private UserDao userDao;

@Override

public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {

User user = userDao.findByUsername(username);

if (null == user) {

LOG.warn("Username {} not found", username);

throw new UsernameNotFoundException("Username " + username + " not found");

}

return user;

}

}

package com.icinbank.onlinebanking.Service.UserServiceImpl;

import com.icinbank.onlinebanking.Entity.User;

import com.icinbank.onlinebanking.Repository.RoleDao;

import com.icinbank.onlinebanking.Repository.UserDao;

import com.icinbank.onlinebanking.Security.UserRole;

import com.icinbank.onlinebanking.Service.AccountService;

import com.icinbank.onlinebanking.Service.UserService;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder;

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import java.util.List;

import java.util.Set;

@Service

@Transactional

public class UserServiceImpl implements UserService {

private static final Logger LOG = LoggerFactory.getLogger(UserService.class);

@Autowired

private UserDao userDao;

@Autowired

private RoleDao roleDao;

@Autowired

private BCryptPasswordEncoder passwordEncoder;

@Autowired

private AccountService accountService;

public void save(User user) {

userDao.save(user);

}

public User findByUsername(String username) {

return userDao.findByUsername(username);

}

public User findByEmail(String email) {

return userDao.findByEmail(email);

}

public User createUser(User user, Set<UserRole> userRoles) {

User localUser = userDao.findByUsername(user.getUsername());

if (localUser != null) {

LOG.info("User with username {} already exist. Nothing will be done. ", user.getUsername());

} else {

String encryptedPassword = passwordEncoder.encode(user.getPassword());

user.setPassword(encryptedPassword);

for (UserRole ur : userRoles) {

roleDao.save(ur.getRole());

}

user.getUserRoles().addAll(userRoles);

user.setPrimaryAccount(accountService.createPrimaryAccount());

user.setSavingsAccount(accountService.createSavingsAccount());

localUser = userDao.save(user);

}

return localUser;

}

public boolean checkUserExists(String username, String email) {

return checkUsernameExists(username) || checkEmailExists(username);

}

public boolean checkUsernameExists(String username) {

return null != findByUsername(username);

}

public boolean checkEmailExists(String email) {

return null != findByEmail(email);

}

public User saveUser(User user) {

return userDao.save(user);

}

public List<User> findUserList() {

return userDao.findAll();

}

public void enableUser(String username) {

User user = findByUsername(username);

user.setEnabled(true);

userDao.save(user);

}

public void disableUser(String username) {

User user = findByUsername(username);

user.setEnabled(false);

LOG.info("user status : {}", user.isEnabled());

userDao.save(user);

LOG.info(username, "{}, is disabled.");

}

}

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:fragment="common-header">

<meta charset="utf-8"/>

<meta http-equiv="X-UA-Compatible" content="IE=edge"/>

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

<meta name="description" content=""/>

<meta name="author" content=""/>

<title>User Front</title>

<!-- Bootstrap Core CSS -->

<link href="/css/bootstrap.min.css" rel="stylesheet"/>

<!-- Website Font style -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.1/css/font-awesome.min.css"/>

<!-- Google Fonts -->

<link href='https://fonts.googleapis.com/css?family=Passion+One' rel='stylesheet' type='text/css'/>

<link href='https://fonts.googleapis.com/css?family=Oxygen' rel='stylesheet' type='text/css'/>

<!--bootstrap date-time picker-->

<link href="/css/bootstrap-datetimepicker.min.css" rel="stylesheet"/>

<!--data table-->

<link href="/css/jquery.dataTables.min.css" rel="stylesheet"/>

<link href="/css/dataTables.bootstrap.min.css" rel="stylesheet"/>

<!-- Custom CSS -->

<!--<link href="css/scrolling-nav.css" rel="stylesheet" />-->

<link href="/css/main.css" rel="stylesheet"/>

</head>

<body>

<!-- Navigation -->

<div th:fragment="navbar">

<img alt="banner" class="img-responsive" src="/images/banner2.png"/>

<nav class="navbar navbar-default " role="navigation">

<div class="container">

<div class="navbar-header page-scroll">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-ex1-collapse">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand page-scroll" th:href="@{/userFront}">Home</a>

</div>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse navbar-ex1-collapse">

<ul class="nav navbar-nav">

<!-- Hidden li included to remove active class from about link when scrolled up past about section -->

<li class="hidden">

<a class="page-scroll" href="#page-top"></a>

</li>

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"

aria-expanded="false">Accounts <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a th:href="@{/account/primaryAccount}">Primary</a></li>

<li role="separator" class="divider"></li>

<li><a th:href="@{/account/savingsAccount}">Savings</a></li>

</ul>

</li>

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"

aria-expanded="false">Transfer <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a th:href="@{/transfer/betweenAccounts}">Between Accounts</a></li>

<li><a th:href="@{/transfer/toSomeoneElse}">To Someone Else</a></li>

<li role="separator" class="divider"></li>

<li><a th:href="@{/transfer/recipient}">Add/Edit Recipient</a></li>

</ul>

</li>

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"

aria-expanded="false">Appointment <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a th:href="@{/appointment/create}">Schedule an Appointment</a></li>

</ul>

</li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li class="dropdown">

<a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"

aria-expanded="false">Me <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a th:href="@{/user/profile}">Profile</a></li>

<li role="separator" class="divider"></li>

<li><a th:href="@{/logout}">Logout</a></li>

</ul>

</li>

</ul>

</div>

<!-- /.navbar-collapse -->

</div>

<!-- /.container -->

</nav>

</div>

<div th:fragment="body-bottom-scripts">

<!-- jQuery -->

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

<!-- Bootstrap Core JavaScript -->

<script src="/js/bootstrap.min.js"></script>

<script src="/js/jquery.easing.min.js"></script>

<!--bootstrap date-time picker-->

<script src="/js/bootstrap-datetimepicker.min.js"></script>

<!--bootbox-->

<script src="/js/bootbox.min.js"></script>

<!--data table-->

<script src="/js/jquery.dataTables.min.js"></script>

<script src="/js/dataTables.bootstrap.min.js"></script>

<!--main js-->

<script src="/js/main.js"></script>

</div>

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>$Title$</title>

</head>

<body>

$END$

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<h3>Schedule An Appointment</h3>

<form th:action="@{/appointment/create}" method="post" id="appointmentForm">

<input type="hidden" name="id" th:value="${appointment.id}"/>

<div class="form-group">

<label for="dateString">Pick a Date and Time:</label>

<div class="input-append date form\_datetime input-group" data-date="2017-01-20T15:25:00Z">

<input class="form-control" type="text" value="" readonly="readonly"

th:value="${dateString}" name="dateString"

id="dateString" required="required"/>

<span class="input-group-addon"><i class="fa fa-times" aria-hidden="true"></i></span>

<span class="input-group-addon"><i class="fa fa-calendar" aria-hidden="true"></i></span>

</div>

</div>

<div class="form-group">

<label for="location">Pick a location:</label>

<select class="form-control" th:value="${appointment.location}" name="location" id="location"

>

<option disabled="disabled" selected="selected"> -- select the location --</option>

<option>Helsinki</option>

<option>Turku</option>

<option>Tampere</option>

<option>Oulu</option>

</select>

</div>

<div class="form-group">

<label for="description">Notes:</label>

<textarea th:value="${appointment.description}" class="form-control" name="description"

id="description" placeholder="Description"></textarea>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<a class="btn btn-primary" id="submitAppointment">Submit Scheduling</a>

</form>

<div class="col-md-6">

</div>

</div>

</div>

<br/>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<form th:action="@{/transfer/betweenAccounts}" method="post">

<div class="form-group">

<label for="transferFrom">1. Please select the account you would like to transfer From:</label>

<select class="form-control" th:value="${transferFrom}" name="transferFrom" id="transferFrom"

required="required">

<option disabled="disabled" selected="selected"> -- select your account --</option>

<option value="Primary">Primary</option>

<option value="Savings">Savings</option>

</select>

<br/>

</div>

<div class="form-group">

<label for="transferTo">2. Please select the account you would like to transfer To:</label>

<select class="form-control" th:value="${transferTo}" name="transferTo" id="transferTo"

required="required">

<option disabled="disabled" selected="selected"> -- select your account --</option>

<option value="Primary">Primary</option>

<option value="Savings">Savings</option>

</select>

<br/>

</div>

<div class="form-group">

<label>3. Please specify the amount you would like to transfer: </label>

<span class="input-group-addon">Amount $</span>

<input th:value="${amount}" type="text" name="amount" id="amount" class="form-control"

aria-label="Amount (to the nearest dollar)"/>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<button class="btn btn-lg btn-primary btn-block" type="submit">Transfer</button>

</form>

<div class="col-md-6">

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<form th:action="@{/account/deposit}" method="post">

<div class="form-group">

<label for="accountType">1. Please select the account you would like to deposit:</label>

<select class="form-control" th:value="${accountType}" name="accountType" id="accountType"

required="required">

<option disabled="disabled" selected="selected"> -- select your account --</option>

<option>Primary</option>

<option>Savings</option>

</select>

<br/>

</div>

<div class="form-group">

<label>2. Please specify the amount you would like to deposit: </label>

<span class="input-group-addon">Amount $</span>

<input th:value="${amount}" type="text" name="amount" id="amount" class="form-control"

aria-label="Amount (to the nearest dollar)"/>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<button class="btn btn-lg btn-primary btn-block" type="submit">Deposit</button>

</form>

<div class="col-md-6">

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<style>

img {

display: block;

margin-left: auto;

margin-right: auto;

}

</style>

<body>

<!-- Login Section -->

<img class="img-responsive" src="/images/banner.png" alt="banner"/>

<div class="container">

<div class="row ">

<div class="main-center ">

<div class="bg-danger" th:if="${param.error}">

Invalid username and secret.

</div>

<div class="bg-danger" th:if="${param.logout}">

You have been logged out.

</div>

<form class="form-signin" th:action="@{/index}" method="post">

<h2 class="form-signin-heading">Please sign in</h2>

<div class="form-group">

<label for="username" class="sr-only">Username</label>

<input type="text" roleId="username" class="form-control" placeholder="Username" name="username"

id="username"

required="required" autofocus="autofocus"/>

</div>

<div class="form-group">

<label for="password" class="sr-only">Password</label>

<input type="password" roleId="inputPassword" class="form-control" placeholder="Password"

id="password"

name="password" required="required"/>

</div>

<div class="form-group">

<input type="checkbox" name="remember-me" id="remember-me"/> &nbsp; Remember me

</div>

<button class="btn btn-lg btn-primary btn-block" type="submit">Sign in</button>

</form>

<hr/>

<div class="form-group ">

<a type="submit" class="btn btn-info btn-lg btn-block login-button" th:href="@{/signup}">Sign up!</a>

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-lg-12">

<!--<h1 class="page-header">Dashboard</h1>-->

</div>

<!-- /.col-lg-12 -->

</div>

<!-- /.row -->

<div class="row">

<div class="col-lg-6 col-md-6">

</div>

<div class="col-lg-6 col-md-6">

<div class="panel panel-info">

<div class="panel-heading">

<div class="row">

<div class="col-xs-6">

<h2>Primary Balance: </h2>

</div>

<div class="col-xs-6 text-right">

<h1><i class="fa fa-usd" aria-hidden="true"></i> <span

th:text="${primaryAccount.accountBalance}">...</span></h1>

</div>

</div>

</div>

<div class="panel-footer">

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</div>

</div>

</div>

<!-- /.row -->

<div class="table-responsive">

<table id="example" class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th>Post Date</th>

<th>Description</th>

<th>Type</th>

<th>Status</th>

<th>Amount</th>

<th>Available Balance</th>

</tr>

</thead>

<tbody>

<tr data-th-each="primaryTransaction : ${primaryTransactionList}">

<td data-th-text="${primaryTransaction.date}">...</td>

<td data-th-text="${primaryTransaction.description}">...</td>

<td data-th-text="${primaryTransaction.type}">...</td>

<td data-th-text="${primaryTransaction.status}">...</td>

<td data-th-text="${primaryTransaction.amount}">...</td>

<td data-th-text="${primaryTransaction.availableBalance}">...</td>

</tr>

</tbody>

</table>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

<script>

$(document).ready(function () {

$('#example').DataTable();

});

</script>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-lg-12">

<!--<h1 class="page-header">Dashboard</h1>-->

</div>

<!-- /.col-lg-12 -->

</div>

<!-- /.row -->

<div class="row">

<div class="col-lg-6 col-md-6">

</div>

<div class="col-lg-6 col-md-6">

<div class="panel panel-success">

<div class="panel-heading">

<div class="row">

<div class="col-xs-6">

<h2>Savings Balance: </h2>

</div>

<div class="col-xs-6 text-right">

<h1><i class="fa fa-usd" aria-hidden="true"></i> <span

th:text="${savingsAccount.accountBalance}">...</span></h1>

</div>

</div>

</div>

<div class="panel-footer">

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</div>

</div>

</div>

<!-- /.row -->

<div class="table-responsive">

<table class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th>Post Date</th>

<th>Description</th>

<th>Type</th>

<th>Status</th>

<th>Amount</th>

<th>Available Balance</th>

</tr>

</thead>

<tbody>

<tr data-th-each="savingsTransaction : ${savingsTransactionList}">

<td data-th-text="${savingsTransaction.date}">...</td>

<td data-th-text="${savingsTransaction.description}">...</td>

<td data-th-text="${savingsTransaction.type}">...</td>

<td data-th-text="${savingsTransaction.status}">...</td>

<td data-th-text="${savingsTransaction.amount}">...</td>

<td data-th-text="${savingsTransaction.availableBalance}">...</td>

</tr>

</tbody>

</table>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<form th:action="@{/transfer/toSomeoneElse}" method="post">

<div class="form-group">

<label for="recipientName">1. Please choose the recipient:</label>

<select class="form-control" th:value="${recipientName}" name="recipientName" id="recipientName"

required="required">

<option disabled="disabled" selected="selected"> -- select the recipient --</option>

<option data-th-each="eachRecipient : ${recipientList}"

data-th-text="${eachRecipient.name}"></option>

</select>

<br/>

</div>

<div class="form-group">

<label for="accountType">2. Please select the account you would like to transfer from:</label>

<select class="form-control" th:value="${accountType}" name="accountType" id="accountType"

required="required">

<option disabled="disabled" selected="selected"> -- select your account --</option>

<option>Primary</option>

<option>Savings</option>

</select>

<br/>

</div>

<div class="form-group">

<label>3. Please specify the amount you would like to transfer: </label>

<span class="input-group-addon">Amount $</span>

<input th:value="${amount}" type="text" name="amount" id="amount" class="form-control"

aria-label="Amount (to the nearest dollar)"/>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<button class="btn btn-lg btn-primary btn-block" type="submit">Transfer</button>

</form>

<div class="col-md-6">

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container">

<div class="row main">

<div class="col-md-6">

<h2 class="title">My Profile</h2>

<hr/>

<form method="post" th:action="@{/user/profile}">

<input type="hidden" name="id" th:value="${user.userId}"/>

<div class="form-group">

<label for="firstName">First Name</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-user fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.firstName}" id="firstName"

name="firstName" roleId="firstname" placeholder="Enter your first name"

required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="lastName" class="cols-sm-2 control-label">Last Name</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-user fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.lastName}" id="lastName"

name="lastName" roleId="lastName" placeholder="Enter your last name"

required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="phone" class="cols-sm-2 control-label">Phone</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-phone fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.phone}" id="phone" name="phone"

roleId="phone" placeholder="xxx-xxx-xxxx" required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="email" class="cols-sm-2 control-label">Your Email</label><span

class="bg-danger pull-right" th:if="${emailExists}">Email already exists</span>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-envelope fa"

aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.email}" id="email" name="email"

roleId="email" placeholder="Enter your Email" required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="username" class="cols-sm-2 control-label">Username</label><span

class="bg-danger pull-right" th:if="${usernameExists}">Username already exists</span>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-users fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.username}" id="username"

name="username" roleId="username" placeholder="Enter your Username"

required="required"/>

</div>

</div>

</div>

<div class="form-group ">

<button type="submit" class="btn btn-primary btn-block ">Change Settings

</button>

</div>

</form>

</div>

<div class="col-md-6">

<div class="panel-heading">

<div class="panel-title text-center">

<h1 class="title">Your Account Information</h1>

<hr/>

</div>

<table class="responstable">

<tr>

<th>Primary Account Number</th>

<th>Savings Account Number</th>

</tr>

<tr>

<td data-th-text="${user.primaryAccount.accountNumber}">...</td>

<td data-th-text="${user.savingsAccount.accountNumber}">...</td>

</tr>

</table>

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<h3>Recipient Information</h3>

<form th:action="@{/transfer/recipient/save}" method="post">

<input type="hidden" name="id" th:value="${recipient.id}"/>

<div class="form-group">

<label for="recipientName">Name</label>

<input type="text" class="form-control" th:value="${recipient.name}" name="name"

id="recipientName" placeholder="Name"/>

</div>

<div class="form-group">

<label for="recipientEmail">Email</label>

<input type="email" class="form-control" th:value="${recipient.email}" name="email"

id="recipientEmail" placeholder="Email"/>

</div>

<div class="form-group">

<label for="recipientPhone">Phone</label>

<input type="text" class="form-control" th:value="${recipient.phone}" name="phone"

id="recipientPhone" placeholder="Phone"/>

</div>

<div class="form-group">

<label for="recipientAccountNumber">Account Number</label>

<input type="text" class="form-control" th:value="${recipient.accountNumber}"

name="accountNumber" id="recipientAccountNumber" placeholder="Account Number"/>

</div>

<div class="form-group">

<label for="recipientDescription">Description</label>

<textarea class="form-control" th:value="${recipient.description}" name="description"

id="recipientDescription" placeholder="Description"

th:text="${recipient.description}"></textarea>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<button class="btn btn-primary" type="submit">Add/Edit Recipient</button>

</form>

<div class="col-md-6">

</div>

</div>

</div>

<br/>

<div class="row">

<h3>List of Recipients</h3>

<div class="table-responsive">

<table class="table table-bordered table-hover table-striped">

<thead>

<tr>

<th>Recipient Name</th>

<th>Recipient Email</th>

<th>Recipient Phone</th>

<th>Recipient Account Number</th>

<th>Description</th>

<th></th>

</tr>

</thead>

<tbody>

<tr data-th-each="eachRecipient : ${recipientList}">

<td><a data-th-text="${eachRecipient.name}"

th:href="@{/transfer/recipient/edit?recipientName=}+${eachRecipient.name}">...</a></td>

<td data-th-text="${eachRecipient.email}">...</td>

<td data-th-text="${eachRecipient.phone}">...</td>

<td data-th-text="${eachRecipient.accountNumber}">...</td>

<td data-th-text="${eachRecipient.description}">...</td>

<td><a th:href="@{/transfer/recipient/delete?recipientName=}+${eachRecipient.name}">delete</a></td>

</tr>

</tbody>

</table>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<!--<head th:replace="common/header :: navbar" />-->

<div class="container">

<div class="row main">

<div class="panel-heading">

<div class="panel-title text-center">

<h1 class="title">Sign Up</h1>

<hr/>

</div>

</div>

<div class="main-login main-center">

<form class="form-horizontal" method="post" th:action="@{/signup}">

<div class="form-group">

<label for="firstName" class="cols-sm-2 control-label">First Name</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-user fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.firstName}" id="firstName"

name="firstName" roleId="firstname" placeholder="Enter your first name"

required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="lastName" class="cols-sm-2 control-label">Last Name</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-user fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.lastName}" id="lastName"

name="lastName" roleId="lastName" placeholder="Enter your last name"

required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="phone" class="cols-sm-2 control-label">Phone</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-phone fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.phone}" id="phone" name="phone"

roleId="phone" placeholder="xxx-xxx-xxxx" required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="email" class="cols-sm-2 control-label">Your Email</label><span

class="bg-danger pull-right" th:if="${emailExists}">Email already exists</span>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-envelope fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.email}" id="email" name="email"

roleId="email" placeholder="Enter your Email" required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="username" class="cols-sm-2 control-label">Username</label><span

class="bg-danger pull-right" th:if="${usernameExists}">Username already exists</span>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-users fa" aria-hidden="true"></i></span>

<input type="text" class="form-control" th:value="${user.username}" id="username"

name="username" roleId="username" placeholder="Enter your Username"

required="required"/>

</div>

</div>

</div>

<div class="form-group">

<label for="password" class="cols-sm-2 control-label">Password</label>

<div class="cols-sm-10">

<div class="input-group">

<span class="input-group-addon"><i class="fa fa-lock fa-lg" aria-hidden="true"></i></span>

<input type="password" class="form-control" th:value="${user.password}" id="password"

name="password" roleId="password" placeholder="Enter your Password"

required="required"/>

</div>

<div class="checkbox">

<label>

<input id="showPassword" type="checkbox"/>Show Password

</label>

</div>

</div>

</div>

<!--<div class="form-group">-->

<!--<label for="confirm" class="cols-sm-2 control-label">Confirm Secret</label>-->

<!--<div class="cols-sm-10">-->

<!--<div class="input-group">-->

<!--<span class="input-group-addon"><i class="fa fa-lock fa-lg" aria-hidden="true"></i></span>-->

<!--<input type="secret" class="form-control" name="confirm" roleId="confirm" placeholder="Confirm your Secret" required="required"/>-->

<!--</div>-->

<!--</div>-->

<!--</div>-->

<div class="form-group ">

<button type="submit" class="btn btn-primary btn-lg btn-block login-button">Sign up!</button>

</div>

<hr/>

<div class="form-group ">

<a class="btn btn-info btn-lg btn-block login-button" th:href="@{/}">Cancel</a>

</div>

</form>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-lg-12">

</div>

<!-- /.col-lg-12 -->

</div>

<!-- /.row -->

<div class="row">

<div class="col-lg-6 col-md-6">

<div class="panel panel-info">

<div class="panel-heading">

<div class="row">

<div class="col-xs-6">

<!--<i class="fa fa-comments fa-5x"></i>-->

<h2>Primary Balance: </h2>

</div>

<div class="col-xs-6 text-right">

<h1><i class="fa fa-usd" aria-hidden="true"></i> <span

th:text="${primaryAccount.accountBalance}">...</span></h1>

</div>

</div>

</div>

<a th:href="@{/account/primaryAccount}">

<div class="panel-footer">

<span class="pull-left">View Details</span>

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</a>

</div>

<div class="panel panel-success">

<div class="panel-heading">

<div class="row">

<div class="col-xs-6">

<h2>Savings Balance: </h2>

</div>

<div class="col-xs-6 text-right">

<h1><i class="fa fa-usd" aria-hidden="true"></i> <span

th:text="${savingsAccount.accountBalance}">...</span></h1>

</div>

</div>

</div>

<a th:href="@{/account/savingsAccount}">

<div class="panel-footer">

<span class="pull-left">View Details</span>

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</a>

</div>

</div>

<div class="col-lg-3 col-md-6">

<div class="panel panel-yellow">

<div class="panel-heading">

<div class="row">

<div class="col-xs-3">

<i class="fa fa-credit-card fa-5x"></i>

</div>

<div class="col-xs-9 text-right">

<div>Deposit</div>

</div>

</div>

</div>

<a th:href="@{/account/deposit}">

<div class="panel-footer">

<span class="pull-left">Go to Deposit</span>

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</a>

</div>

</div>

<div class="col-lg-3 col-md-6">

<div class="panel panel-red">

<div class="panel-heading">

<div class="row">

<div class="col-xs-3">

<i class="fa fa-money fa-5x"></i>

</div>

<div class="col-xs-9 text-right">

<div>Withdraw</div>

</div>

</div>

</div>

<a th:href="@{/account/withdraw}">

<div class="panel-footer">

<span class="pull-left">Go to Withdrawal</span>

<span class="pull-right"><i class="fa fa-arrow-circle-right"></i></span>

<div class="clearfix"></div>

</div>

</a>

</div>

</div>

</div>

<!-- /.row -->

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

<!DOCTYPE html>

<html xmlns:th="http://www.w3.org/1999/xhtml" lang="en">

<head th:replace="common/header :: common-header"/>

<body roleId="page-top" data-spy="scroll" data-target=".navbar-fixed-top">

<head th:replace="common/header :: navbar"/>

<div class="container main">

<div class="row">

<div class="col-md-6">

<form th:action="@{/account/withdraw}" method="post">

<div class="form-group">

<label for="accountType">1. Please select the account you would like to withdraw:</label>

<select class="form-control" th:value="${accountType}" name="accountType" id="accountType"

required="required">

<option disabled="disabled" selected="selected"> -- select your account --</option>

<option>Primary</option>

<option>Savings</option>

</select>

<br/>

</div>

<div class="form-group">

<label>2. Please specify the amount you would like to withdraw: </label>

<span class="input-group-addon">Amount $</span>

<input th:value="${amount}" type="text" name="amount" id="amount" class="form-control"

aria-label="Amount (to the nearest dollar)"/>

</div>

<input type="hidden"

name="${\_csrf.parameterName}"

value="${\_csrf.token}"/>

<button class="btn btn-lg btn-primary btn-block" type="submit">Withdraw</button>

</form>

<div class="col-md-6">

</div>

</div>

</div>

</div>

<div th:replace="common/header :: body-bottom-scripts"/>

</body>

</html>

/\*!

\* Bootstrap v3.3.7 ([http://getbootstrap.com](http://getbootstrap.com/))

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\*/

/\*! normalize.css v3.0.3 | MIT License | github.com/necolas/normalize.css \*/

html {

font-family: sans-serif;

-webkit-text-size-adjust: 100%;

-ms-text-size-adjust: 100%;

}

body {

margin: 0;

}

article,

aside,

details,

figcaption,

figure,

footer,

header,

hgroup,

main,

menu,

nav,

section,

summary {

display: block;

}

audio,

canvas,

progress,

video {

display: inline-block;

vertical-align: baseline;

}

audio:not([controls]) {

display: none;

height: 0;

}

[hidden],

template {

display: none;

}

a {

background-color: transparent;

}

a:active,

a:hover {

outline: 0;

}

abbr[title] {

border-bottom: 1px dotted;

}

b,

strong {

font-weight: bold;

}

dfn {

font-style: italic;

}

h1 {

margin: .67em 0;

font-size: 2em;

}

mark {

color: #000;

background: #ff0;

}

small {

font-size: 80%;

}

sub,

sup {

position: relative;

font-size: 75%;

line-height: 0;

vertical-align: baseline;

}

sup {

top: -.5em;

}

sub {

bottom: -.25em;

}

img {

border: 0;

}

svg:not(:root) {

overflow: hidden;

}

figure {

margin: 1em 40px;

}

hr {

height: 0;

-webkit-box-sizing: content-box;

-moz-box-sizing: content-box;

box-sizing: content-box;

}

pre {

overflow: auto;

}

code,

kbd,

pre,

samp {

font-family: monospace, monospace;

font-size: 1em;

}

button,

input,

optgroup,

select,

textarea {

margin: 0;

font: inherit;

color: inherit;

}

button {

overflow: visible;

}

button,

select {

text-transform: none;

}

button,

html input[type="button"],

input[type="reset"],

input[type="submit"] {

-webkit-appearance: button;

cursor: pointer;

}

button[disabled],

html input[disabled] {

cursor: default;

}

button::-moz-focus-inner,

input::-moz-focus-inner {

padding: 0;

border: 0;

}

input {

line-height: normal;

}

input[type="checkbox"],

input[type="radio"] {

-webkit-box-sizing: border-box;

-moz-box-sizing: border-box;

box-sizing: border-box;

padding: 0;

}

input[type="number"]::-webkit-inner-spin-button,

input[type="number"]::-webkit-outer-spin-button {

height: auto;

}

input[type="search"] {

-webkit-box-sizing: content-box;

-moz-box-sizing: content-box;

box-sizing: content-box;

-webkit-appearance: textfield;

}

input[type="search"]::-webkit-search-cancel-button,

input[type="search"]::-webkit-search-decoration {

-webkit-appearance: none;

}

fieldset {

padding: .35em .625em .75em;

margin: 0 2px;

border: 1px solid #c0c0c0;

}

legend {

padding: 0;

border: 0;

}

textarea {

overflow: auto;

}

optgroup {

font-weight: bold;

}

table {

border-spacing: 0;

border-collapse: collapse;

}

td,

th {

padding: 0;

}

/\*! Source: <https://github.com/h5bp/html5-boilerplate/blob/master/src/css/main.css> \*/

@media print {

\*,

\*:before,

\*:after {

color: #000 !important;

text-shadow: none !important;

background: transparent !important;

-webkit-box-shadow: none !important;

box-shadow: none !important;

}

a,

a:visited {

text-decoration: underline;

}

a[href]:after {

content: " (" attr(href) ")";

}

abbr[title]:after {

content: " (" attr(title) ")";

}

a[href^="#"]:after,

a[href^="javascript:"]:after {

content: "";

}

pre,

blockquote {

border: 1px solid #999;

page-break-inside: avoid;

}

thead {

display: table-header-group;

}

tr,

img {

page-break-inside: avoid;

}

img {

max-width: 100% !important;

}

p,

h2,

h3 {

orphans: 3;

widows: 3;

}

h2,

h3 {

page-break-after: avoid;

}

.navbar {

display: none;

}

.btn > .caret,

.dropup > .btn > .caret {

border-top-color: #000 !important;

}

.label {

border: 1px solid #000;

}

.table {

border-collapse: collapse !important;

}

.table td,

.table th {

background-color: #fff !important;

}

.table-bordered th,

.table-bordered td {

border: 1px solid #ddd !important;

}

}

@font-face {

font-family: 'Glyphicons Halflings';

src: url('../fonts/glyphicons-halflings-regular.eot');

src: url('../fonts/glyphicons-halflings-regular.eot?#iefix') format('embedded-opentype'), url('../fonts/glyphicons-halflings-regular.woff2') format('woff2'), url('../fonts/glyphicons-halflings-regular.woff') format('woff'), url('../fonts/glyphicons-halflings-regular.ttf') format('truetype'), url('../fonts/glyphicons-halflings-regular.svg#glyphicons\_halflingsregular') format('svg');

}

.glyphicon {

position: relative;

top: 1px;

display: inline-block;

font-family: 'Glyphicons Halflings';

font-style: normal;

font-weight: normal;

line-height: 1;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

.glyphicon-asterisk:before {

content: "\002a";

}

.glyphicon-plus:before {

content: "\002b";

}

.glyphicon-euro:before,

.glyphicon-eur:before {

content: "\20ac";

}

.glyphicon-minus:before {

content: "\2212";

}

.glyphicon-cloud:before {

content: "\2601";

}

.glyphicon-envelope:before {

content: "\2709";

}

.glyphicon-pencil:before {

content: "\270f";

}

.glyphicon-glass:before {

content: "\e001";

}

.glyphicon-music:before {

content: "\e002";

}

.glyphicon-search:before {

content: "\e003";

}

.glyphicon-heart:before {

content: "\e005";

}

.glyphicon-star:before {

content: "\e006";

}

.glyphicon-star-empty:before {

content: "\e007";

}

.glyphicon-user:before {

content: "\e008";

}

.glyphicon-film:before {

content: "\e009";

}

.glyphicon-th-large:before {

content: "\e010";

}

.glyphicon-th:before {

content: "\e011";

}

.glyphicon-th-list:before {

content: "\e012";

}

.glyphicon-ok:before {

content: "\e013";

}

.glyphicon-remove:before {

content: "\e014";

}

.glyphicon-zoom-in:before {

content: "\e015";

}

.glyphicon-zoom-out:before {

content: "\e016";

}

.glyphicon-off:before {

content: "\e017";

}

.glyphicon-signal:before {

content: "\e018";

}

.glyphicon-cog:before {

content: "\e019";

}

.glyphicon-trash:before {

content: "\e020";

}

.glyphicon-home:before {

content: "\e021";

}

.glyphicon-file:before {

content: "\e022";

}

.glyphicon-time:before {

content: "\e023";

}

.glyphicon-road:before {

content: "\e024";

}

.glyphicon-download-alt:before {

content: "\e025";

}

.glyphicon-download:before {

content: "\e026";

}

.glyphicon-upload:before {

content: "\e027";

}

.glyphicon-inbox:before {

content: "\e028";

}

.glyphicon-play-circle:before {

content: "\e029";

}

.glyphicon-repeat:before {

content: "\e030";

}

.glyphicon-refresh:before {

content: "\e031";

}

.glyphicon-list-alt:before {

content: "\e032";

}

.glyphicon-lock:before {

content: "\e033";

}

.glyphicon-flag:before {

content: "\e034";

}

.glyphicon-headphones:before {

content: "\e035";

}

.glyphicon-volume-off:before {

content: "\e036";

}

.glyphicon-volume-down:before {

content: "\e037";

}

.glyphicon-volume-up:before {

content: "\e038";

}

.glyphicon-qrcode:before {

content: "\e039";

}

.glyphicon-barcode:before {

content: "\e040";

}

.glyphicon-tag:before {

content: "\e041";

}

.glyphicon-tags:before {

content: "\e042";

}

.glyphicon-book:before {

content: "\e043";

}

.glyphicon-bookmark:before {

content: "\e044";

}

.glyphicon-print:before {

content: "\e045";

}

.glyphicon-camera:before {

content: "\e046";

}

.glyphicon-font:before {

content: "\e047";

}

.glyphicon-bold:before {

content: "\e048";

}

.glyphicon-italic:before {

content: "\e049";

}

.glyphicon-text-height:before {

content: "\e050";

}

.glyphicon-text-width:before {

content: "\e051";

}

.glyphicon-align-left:before {

content: "\e052";

}

.glyphicon-align-center:before {

content: "\e053";

}

.glyphicon-align-right:before {

content: "\e054";

}

.glyphicon-align-justify:before {

content: "\e055";

}

.glyphicon-list:before {

content: "\e056";

}

.glyphicon-indent-left:before {

content: "\e057";

}

.glyphicon-indent-right:before {

content: "\e058";

}

.glyphicon-facetime-video:before {

content: "\e059";

}

.glyphicon-picture:before {

content: "\e060";

}

.glyphicon-map-marker:before {

content: "\e062";

}

.glyphicon-adjust:before {

content: "\e063";

}

.glyphicon-tint:before {

content: "\e064";

}

.glyphicon-edit:before {

content: "\e065";

}

.glyphicon-share:before {

content: "\e066";

}

.glyphicon-check:before {

content: "\e067";

}

.glyphicon-move:before {

content: "\e068";

}

.glyphicon-step-backward:before {

content: "\e069";

}

.glyphicon-fast-backward:before {

content: "\e070";

}

.glyphicon-backward:before {

content: "\e071";

}

.glyphicon-play:before {

content: "\e072";

}

.glyphicon-pause:before {

content: "\e073";

}

.glyphicon-stop:before {

content: "\e074";

}

.glyphicon-forward:before {

content: "\e075";

}

.glyphicon-fast-forward:before {

content: "\e076";

}

.glyphicon-step-forward:before {

content: "\e077";

}

.glyphicon-eject:before {

content: "\e078";

}

.glyphicon-chevron-left:before {

content: "\e079";

}

.glyphicon-chevron-right:before {

content: "\e080";

}

.glyphicon-plus-sign:before {

content: "\e081";

}

.glyphicon-minus-sign:before {

content: "\e082";

}

.glyphicon-remove-sign:before {

content: "\e083";

}

.glyphicon-ok-sign:before {

content: "\e084";

}

.glyphicon-question-sign:before {

content: "\e085";

}

.glyphicon-info-sign:before {

content: "\e086";

}

.glyphicon-screenshot:before {

content: "\e087";

}

.glyphicon-remove-circle:before {

content: "\e088";

}

.glyphicon-ok-circle:before {

content: "\e089";

}

.glyphicon-ban-circle:before {

content: "\e090";

}

.glyphicon-arrow-left:before {

content: "\e091";

}

.glyphicon-arrow-right:before {

content: "\e092";

}

.glyphicon-arrow-up:before {

content: "\e093";

}

.glyphicon-arrow-down:before {

content: "\e094";

}

.glyphicon-share-alt:before {

content: "\e095";

}

.glyphicon-resize-full:before {

content: "\e096";

}

.glyphicon-resize-small:before {

content: "\e097";

}

.glyphicon-exclamation-sign:before {

content: "\e101";

}

.glyphicon-gift:before {

content: "\e102";

}

.glyphicon-leaf:before {

content: "\e103";

}

.glyphicon-fire:before {

content: "\e104";

}

.glyphicon-eye-open:before {

content: "\e105";

}

.glyphicon-eye-close:before {

content: "\e106";

}

.glyphicon-warning-sign:before {

content: "\e107";

}

.glyphicon-plane:before {

content: "\e108";

}

.glyphicon-calendar:before {

content: "\e109";

}

.glyphicon-random:before {

content: "\e110";

}

.glyphicon-comment:before {

content: "\e111";

}

.glyphicon-magnet:before {

content: "\e112";

}

.glyphicon-chevron-up:before {

content: "\e113";

}

.glyphicon-chevron-down:before {

content: "\e114";

}

.glyphicon-retweet:before {

content: "\e115";

}

.glyphicon-shopping-cart:before {

content: "\e116";

}

.glyphicon-folder-close:before {

content: "\e117";

}

.glyphicon-folder-open:before {

content: "\e118";

}

.glyphicon-resize-vertical:before {

content: "\e119";

}

.glyphicon-resize-horizontal:before {

content: "\e120";

}

.glyphicon-hdd:before {

content: "\e121";

}

.glyphicon-bullhorn:before {

content: "\e122";

}

.glyphicon-bell:before {

content: "\e123";

}

.glyphicon-certificate:before {

content: "\e124";

}

.glyphicon-thumbs-up:before {

content: "\e125";

}

.glyphicon-thumbs-down:before {

content: "\e126";

}

.glyphicon-hand-right:before {

content: "\e127";

}

.glyphicon-hand-left:before {

content: "\e128";

}

.glyphicon-hand-up:before {

content: "\e129";

}

.glyphicon-hand-down:before {

content: "\e130";

}

.glyphicon-circle-arrow-right:before {

content: "\e131";

}

.glyphicon-circle-arrow-left:before {

content: "\e132";

}

.glyphicon-circle-arrow-up:before {

content: "\e133";

}

.glyphicon-circle-arrow-down:before {

content: "\e134";

}

.glyphicon-globe:before {

content: "\e135";

}

.glyphicon-wrench:before {

content: "\e136";

}

.glyphicon-tasks:before {

content: "\e137";

}

.glyphicon-filter:before {

content: "\e138";

}

.glyphicon-briefcase:before {

content: "\e139";

}

.glyphicon-fullscreen:before {

content: "\e140";

}

.glyphicon-dashboard:before {

content: "\e141";

}

.glyphicon-paperclip:before {

content: "\e142";

}

.glyphicon-heart-empty:before {

content: "\e143";

}

.glyphicon-link:before {

content: "\e144";

}

.glyphicon-phone:before {

content: "\e145";

}

.glyphicon-pushpin:before {

content: "\e146";

}

.glyphicon-usd:before {

content: "\e148";

}

.glyphicon-gbp:before {

content: "\e149";

}

.glyphicon-sort:before {

content: "\e150";

}

.glyphicon-sort-by-alphabet:before {

content: "\e151";

}

.glyphicon-sort-by-alphabet-alt:before {

content: "\e152";

}

.glyphicon-sort-by-order:before {

content: "\e153";

}

.glyphicon-sort-by-order-alt:before {

content: "\e154";

}

.glyphicon-sort-by-attributes:before {

content: "\e155";

}

.glyphicon-sort-by-attributes-alt:before {

content: "\e156";

}

.glyphicon-unchecked:before {

content: "\e157";

}

.glyphicon-expand:before {

content: "\e158";

}

.glyphicon-collapse-down:before {

content: "\e159";

}

.glyphicon-collapse-up:before {

content: "\e160";

}

.glyphicon-log-in:before {

content: "\e161";

}

.glyphicon-flash:before {

content: "\e162";

}

.glyphicon-log-out:before {

content: "\e163";

}

.glyphicon-new-window:before {

content: "\e164";

}

.glyphicon-record:before {

content: "\e165";

}

.glyphicon-save:before {

content: "\e166";

}

.glyphicon-open:before {

content: "\e167";

}

.glyphicon-saved:before {

content: "\e168";

}

.glyphicon-import:before {

content: "\e169";

}

.glyphicon-export:before {

content: "\e170";

}

.glyphicon-send:before {

content: "\e171";

}

.glyphicon-floppy-disk:before {

content: "\e172";

}

.glyphicon-floppy-saved:before {

content: "\e173";

}

.glyphicon-floppy-remove:before {

content: "\e174";

}

.glyphicon-floppy-save:before {

content: "\e175";

}

.glyphicon-floppy-open:before {

content: "\e176";

}

.glyphicon-credit-card:before {

content: "\e177";

}

.glyphicon-transfer:before {

content: "\e178";

}

.glyphicon-cutlery:before {

content: "\e179";

}

.glyphicon-header:before {

content: "\e180";

}

.glyphicon-compressed:before {

content: "\e181";

}

.glyphicon-earphone:before {

content: "\e182";

}

.glyphicon-phone-alt:before {

content: "\e183";

}

.glyphicon-tower:before {

content: "\e184";

}

.glyphicon-stats:before {

content: "\e185";

}

.glyphicon-sd-video:before {

content: "\e186";

}

.glyphicon-hd-video:before {

content: "\e187";

}

.glyphicon-subtitles:before {

content: "\e188";

}

.glyphicon-sound-stereo:before {

content: "\e189";

}

.glyphicon-sound-dolby:before {

content: "\e190";

}

.glyphicon-sound-5-1:before {

content: "\e191";

}

.glyphicon-sound-6-1:before {

content: "\e192";

}

.glyphicon-sound-7-1:before {

content: "\e193";

}

.glyphicon-copyright-mark:before {

content: "\e194";

}

.glyphicon-registration-mark:before {

content: "\e195";

}

.glyphicon-cloud-download:before {

content: "\e197";

}

.glyphicon-cloud-upload:before {

content: "\e198";

}

.glyphicon-tree-conifer:before {

content: "\e199";

}

.glyphicon-tree-deciduous:before {

content: "\e200";

}

.glyphicon-cd:before {

content: "\e201";

}

.glyphicon-save-file:before {

content: "\e202";

}

.glyphicon-open-file:before {

content: "\e203";

}

.glyphicon-level-up:before {

content: "\e204";

}

.glyphicon-copy:before {

content: "\e205";

}

.glyphicon-paste:before {

content: "\e206";

}

.glyphicon-alert:before {

content: "\e209";

}

.glyphicon-equalizer:before {

content: "\e210";

}

.glyphicon-king:before {

content: "\e211";

}

.glyphicon-queen:before {

content: "\e212";

}

.glyphicon-pawn:before {

content: "\e213";

}

.glyphicon-bishop:before {

content: "\e214";

}

.glyphicon-knight:before {

content: "\e215";

}

.glyphicon-baby-formula:before {

content: "\e216";

}

.glyphicon-tent:before {

content: "\26fa";

}

.glyphicon-blackboard:before {

content: "\e218";

}

.glyphicon-bed:before {

content: "\e219";

}

.glyphicon-apple:before {

content: "\f8ff";

}

.glyphicon-erase:before {

content: "\e221";

}

.glyphicon-hourglass:before {

content: "\231b";

}

.glyphicon-lamp:before {

content: "\e223";

}

.glyphicon-duplicate:before {

content: "\e224";

}

.glyphicon-piggy-bank:before {

content: "\e225";

}

.glyphicon-scissors:before {

content: "\e226";

}

.glyphicon-bitcoin:before {

content: "\e227";

}

.glyphicon-btc:before {

content: "\e227";

}

.glyphicon-xbt:before {

content: "\e227";

}

.glyphicon-yen:before {

content: "\00a5";

}

.glyphicon-jpy:before {

content: "\00a5";

}

.glyphicon-ruble:before {

content: "\20bd";

}

.glyphicon-rub:before {

content: "\20bd";

}

.glyphicon-scale:before {

content: "\e230";

}

.glyphicon-ice-lolly:before {

content: "\e231";

}

.glyphicon-ice-lolly-tasted:before {

content: "\e232";

}

.glyphicon-education:before {

content: "\e233";

}

.glyphicon-option-horizontal:before {

content: "\e234";

}

.glyphicon-option-vertical:before {

content: "\e235";

}

.glyphicon-menu-hamburger:before {

content: "\e236";

}

.glyphicon-modal-window:before {

content: "\e237";

}

.glyphicon-oil:before {

content: "\e238";

}

.glyphicon-grain:before {

content: "\e239";

}

.glyphicon-sunglasses:before {

content: "\e240";

}

.glyphicon-text-size:before {

content: "\e241";

}

.glyphicon-text-color:before {

content: "\e242";

}

.glyphicon-text-background:before {

content: "\e243";

}

.glyphicon-object-align-top:before {

content: "\e244";

}

.glyphicon-object-align-bottom:before {

content: "\e245";

}

.glyphicon-object-align-horizontal:before {

content: "\e246";

}

.glyphicon-object-align-left:before {

content: "\e247";

}

.glyphicon-object-align-vertical:before {

content: "\e248";

}

.glyphicon-object-align-right:before {

content: "\e249";

}

.glyphicon-triangle-right:before {

content: "\e250";

}

.glyphicon-triangle-left:before {

content: "\e251";

}

.glyphicon-triangle-bottom:before {

content: "\e252";

}

.glyphicon-triangle-top:before {

content: "\e253";

}

.glyphicon-console:before {

content: "\e254";

}

.glyphicon-superscript:before {

content: "\e255";

}

.glyphicon-subscript:before {

content: "\e256";

}

.glyphicon-menu-left:before {

content: "\e257";

}

.glyphicon-menu-right:before {

content: "\e258";

}

.glyphicon-menu-down:before {

content: "\e259";

}

.glyphicon-menu-up:before {

content: "\e260";

}

\* {

-webkit-box-sizing: border-box;

-moz-box-sizing: border-box;

box-sizing: border-box;

}

\*:before,

\*:after {

-webkit-box-sizing: border-box;

-moz-box-sizing: border-box;

box-sizing: border-box;

}

html {

font-size: 10px;

-webkit-tap-highlight-color: rgba(0, 0, 0, 0);

}

body {

font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;

font-size: 14px;

line-height: 1.42857143;

color: #333;

background-color: #fff;

}

input,

button,

select,

textarea {

font-family: inherit;

font-size: inherit;

line-height: inherit;

}

a {

color: #337ab7;

text-decoration: none;

}

a:hover,

a:focus {

color: #23527c;

text-decoration: underline;

}

a:focus {

outline: 5px auto -webkit-focus-ring-color;

outline-offset: -2px;

}

figure {

margin: 0;

}

img {

vertical-align: middle;

}

.img-responsive,

.thumbnail > img,

.thumbnail a > img,

.carousel-inner > .item > img,

.carousel-inner > .item > a > img {

display: block;

max-width: 100%;

height: auto;

}

.img-rounded {

border-radius: 6px;

}

.img-thumbnail {

display: inline-block;

max-width: 100%;

height: auto;

padding: 4px;

line-height: 1.42857143;

background-color: #fff;

border: 1px solid #ddd;

border-radius: 4px;

-webkit-transition: all .2s ease-in-out;

-o-transition: all .2s ease-in-out;

transition: all .2s ease-in-out;

}

.img-circle {

border-radius: 50%;

}

hr {

margin-top: 20px;

margin-bottom: 20px;

border: 0;

border-top: 1px solid #eee;

}

.sr-only {

position: absolute;

width: 1px;

height: 1px;

padding: 0;

margin: -1px;

overflow: hidden;

clip: rect(0, 0, 0, 0);

border: 0;

}

.sr-only-focusable:active,

.sr-only-focusable:focus {

position: static;

width: auto;

height: auto;

margin: 0;

overflow: visible;

clip: auto;

}

[role="button"] {

cursor: pointer;

}

h1,

h2,

h3,

h4,

h5,

h6,

.h1,

.h2,

.h3,

.h4,

.h5,

.h6 {

font-family: inherit;

font-weight: 500;

line-height: 1.1;

color: inherit;

}

h1 small,

h2 small,

h3 small,

h4 small,

h5 small,

h6 small,

.h1 small,

.h2 small,

.h3 small,

.h4 small,

.h5 small,

.h6 small,

h1 .small,

h2 .small,

h3 .small,

h4 .small,

h5 .small,

h6 .small,

.h1 .small,

.h2 .small,

.h3 .small,

.h4 .small,

.h5 .small,

.h6 .small {

font-weight: normal;

line-height: 1;

color: #777;

}

h1,

.h1,

h2,

.h2,

h3,

.h3 {

margin-top: 20px;

margin-bottom: 10px;

}

h1 small,

.h1 small,

h2 small,

.h2 small,

h3 small,

.h3 small,

h1 .small,

.h1 .small,

h2 .small,

.h2 .small,

h3 .small,

.h3 .small {

font-size: 65%;

}

h4,

.h4,

h5,

.h5,

h6,

.h6 {

margin-top: 10px;

margin-bottom: 10px;

}

h4 small,

.h4 small,

h5 small,

.h5 small,

h6 small,

.h6 small,

h4 .small,

.h4 .small,

h5 .small,

.h5 .small,

h6 .small,

.h6 .small {

font-size: 75%;

}

h1,

.h1 {

font-size: 36px;

}

h2,

.h2 {

font-size: 30px;

}

h3,

.h3 {

font-size: 24px;

}

h4,

.h4 {

font-size: 18px;

}

h5,

.h5 {

font-size: 14px;

}

h6,

.h6 {

font-size: 12px;

}

p {

margin: 0 0 10px;

}

.lead {

margin-bottom: 20px;

font-size: 16px;

font-weight: 300;

line-height: 1.4;

}

@media (min-width: 768px) {

.lead {

font-size: 21px;

}

}

small,

.small {

font-size: 85%;

}

mark,

.mark {

padding: .2em;

background-color: #fcf8e3;

}

.text-left {

text-align: left;

}

.text-right {

text-align: right;

}

.text-center {

text-align: center;

}

.text-justify {

text-align: justify;

}

.text-nowrap {

white-space: nowrap;

}

.text-lowercase {

text-transform: lowercase;

}

.text-uppercase {

text-transform: uppercase;

}

.text-capitalize {

text-transform: capitalize;

}

.text-muted {

color: #777;

}

.text-primary {

color: #337ab7;

}

a.text-primary:hover,

a.text-primary:focus {

color: #286090;

}

.text-success {

color: #3c763d;

}

a.text-success:hover,

a.text-success:focus {

color: #2b542c;

}

.text-info {

color: #31708f;

}

a.text-info:hover,

a.text-info:focus {

color: #245269;

}

.text-warning {

color: #8a6d3b;

}

a.text-warning:hover,

a.text-warning:focus {

color: #66512c;

}

.text-danger {

color: #a94442;

}

a.text-danger:hover,

a.text-danger:focus {

color: #843534;

}

.bg-primary {

color: #fff;

background-color: #337ab7;

}

a.bg-primary:hover,

a.bg-primary:focus {

background-color: #286090;

}

.bg-success {

background-color: #dff0d8;

}

a.bg-success:hover,

a.bg-success:focus {

background-color: #c1e2b3;

}

.bg-info {

background-color: #d9edf7;

}

a.bg-info:hover,

a.bg-info:focus {

background-color: #afd9ee;

}

.bg-warning {

background-color: #fcf8e3;

}

a.bg-warning:hover,

a.bg-warning:focus {

background-color: #f7ecb5;

}

.bg-danger {

background-color: #f2dede;

}

a.bg-danger:hover,

a.bg-danger:focus {

background-color: #e4b9b9;

}

.page-header {

padding-bottom: 9px;

margin: 40px 0 20px;

border-bottom: 1px solid #eee;

}

ul,

ol {

margin-top: 0;

margin-bottom: 10px;

}

ul ul,

ol ul,

ul ol,

ol ol {

margin-bottom: 0;

}

.list-unstyled {

padding-left: 0;

list-style: none;

}

.list-inline {

padding-left: 0;

margin-left: -5px;

list-style: none;

}

.list-inline > li {

display: inline-block;

padding-right: 5px;

padding-left: 5px;

}

dl {

margin-top: 0;

margin-bottom: 20px;

}

dt,

dd {

line-height: 1.42857143;

}

dt {

font-weight: bold;

}

dd {

margin-left: 0;

}

@media (min-width: 768px) {

.dl-horizontal dt {

float: left;

width: 160px;

overflow: hidden;

clear: left;

text-align: right;

text-overflow: ellipsis;

white-space: nowrap;

}

.dl-horizontal dd {

margin-left: 180px;

}

}

abbr[title],

abbr[data-original-title] {

cursor: help;

border-bottom: 1px dotted #777;

}

.initialism {

font-size: 90%;

text-transform: uppercase;

}

blockquote {

padding: 10px 20px;

margin: 0 0 20px;

font-size: 17.5px;

border-left: 5px solid #eee;

}

blockquote p:last-child,

blockquote ul:last-child,

blockquote ol:last-child {

margin-bottom: 0;

}

blockquote footer,

blockquote small,

blockquote .small {

display: block;

font-size: 80%;

line-height: 1.42857143;

color: #777;

}

blockquote footer:before,

blockquote small:before,

blockquote .small:before {

content: '\2014 \00A0';

}

.blockquote-reverse,

blockquote.pull-right {

padding-right: 15px;

padding-left: 0;

text-align: right;

border-right: 5px solid #eee;

border-left: 0;

}

.blockquote-reverse footer:before,

blockquote.pull-right footer:before,

.blockquote-reverse small:before,

blockquote.pull-right small:before,

.blockquote-reverse .small:before,

blockquote.pull-right .small:before {

content: '';

}

.blockquote-reverse footer:after,

blockquote.pull-right footer:after,

.blockquote-reverse small:after,

blockquote.pull-right small:after,

.blockquote-reverse .small:after,

blockquote.pull-right .small:after {

content: '\00A0 \2014';

}

address {

margin-bottom: 20px;

font-style: normal;

line-height: 1.42857143;

}

code,

kbd,

pre,

samp {

font-family: Menlo, Monaco, Consolas, "Courier New", monospace;

}

code {

padding: 2px 4px;

font-size: 90%;

color: #c7254e;

background-color: #f9f2f4;

border-radius: 4px;

}

kbd {

padding: 2px 4px;

font-size: 90%;

color: #fff;

background-color: #333;

border-radius: 3px;

-webkit-box-shadow: inset 0 -1px 0 rgba(0, 0, 0, .25);

box-shadow: inset 0 -1px 0 rgba(0, 0, 0, .25);

}

kbd kbd {

padding: 0;

font-size: 100%;

font-weight: bold;

-webkit-box-shadow: none;

box-shadow: none;

}

pre {

display: block;

padding: 9.5px;

margin: 0 0 10px;

font-size: 13px;

line-height: 1.42857143;

color: #333;

word-break: break-all;

word-wrap: break-word;

background-color: #f5f5f5;

border: 1px solid #ccc;

border-radius: 4px;

}

pre code {

padding: 0;

font-size: inherit;

color: inherit;

white-space: pre-wrap;

background-color: transparent;

border-radius: 0;

}

.pre-scrollable {

max-height: 340px;

overflow-y: scroll;

}

.container {

padding-right: 15px;

padding-left: 15px;

margin-right: auto;

margin-left: auto;

}

@media (min-width: 768px) {

.container {

width: 750px;

}

}

@media (min-width: 992px) {

.container {

width: 970px;

}

}

@media (min-width: 1200px) {

.container {

width: 1170px;

}

}

.container-fluid {

padding-right: 15px;

padding-left: 15px;

margin-right: auto;

margin-left: auto;

}

.row {

margin-right: -15px;

margin-left: -15px;

}

.col-xs-1, .col-sm-1, .col-md-1, .col-lg-1, .col-xs-2, .col-sm-2, .col-md-2, .col-lg-2, .col-xs-3, .col-sm-3, .col-md-3, .col-lg-3, .col-xs-4, .col-sm-4, .col-md-4, .col-lg-4, .col-xs-5, .col-sm-5, .col-md-5, .col-lg-5, .col-xs-6, .col-sm-6, .col-md-6, .col-lg-6, .col-xs-7, .col-sm-7, .col-md-7, .col-lg-7, .col-xs-8, .col-sm-8, .col-md-8, .col-lg-8, .col-xs-9, .col-sm-9, .col-md-9, .col-lg-9, .col-xs-10, .col-sm-10, .col-md-10, .col-lg-10, .col-xs-11, .col-sm-11, .col-md-11, .col-lg-11, .col-xs-12, .col-sm-12, .col-md-12, .col-lg-12 {

position: relative;

min-height: 1px;

padding-right: 15px;

padding-left: 15px;

}

.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, .col-xs-10, .col-xs-11, .col-xs-12 {

float: left;

}

.col-xs-12 {

width: 100%;

}

.col-xs-11 {

width: 91.66666667%;

}

.col-xs-10 {

width: 83.33333333%;

}

.col-xs-9 {

width: 75%;

}

.col-xs-8 {

width: 66.66666667%;

}

.col-xs-7 {

width: 58.33333333%;

}

.col-xs-6 {

width: 50%;

}

.col-xs-5 {

width: 41.66666667%;

}

.col-xs-4 {

width: 33.33333333%;

}

.col-xs-3 {

width: 25%;

}

.col-xs-2 {

width: 16.66666667%;

}

.col-xs-1 {

width: 8.33333333%;

}

.col-xs-pull-12 {

right: 100%;

}

.col-xs-pull-11 {

right: 91.66666667%;

}

.col-xs-pull-10 {

right: 83.33333333%;

}

.col-xs-pull-9 {

right: 75%;

}

.col-xs-pull-8 {

right: 66.66666667%;

}

.col-xs-pull-7 {

right: 58.33333333%;

}

.col-xs-pull-6 {

right: 50%;

}

.col-xs-pull-5 {

right: 41.66666667%;

}

.col-xs-pull-4 {

right: 33.33333333%;

}

.col-xs-pull-3 {

right: 25%;

}

.col-xs-pull-2 {

right: 16.66666667%;

}

.col-xs-pull-1 {

right: 8.33333333%;

}

.col-xs-pull-0 {

right: auto;

}

.col-xs-push-12 {

left: 100%;

}

.col-xs-push-11 {

left: 91.66666667%;

}

.col-xs-push-10 {

left: 83.33333333%;

}

.col-xs-push-9 {

left: 75%;

}

.col-xs-push-8 {

left: 66.66666667%;

}

.col-xs-push-7 {

left: 58.33333333%;

}

.col-xs-push-6 {

left: 50%;

}

.col-xs-push-5 {

left: 41.66666667%;

}

.col-xs-push-4 {

left: 33.33333333%;

}

.col-xs-push-3 {

left: 25%;

}

.col-xs-push-2 {

left: 16.66666667%;

}

.col-xs-push-1 {

left: 8.33333333%;

}

.col-xs-push-0 {

left: auto;

}

.col-xs-offset-12 {

margin-left: 100%;

}

.col-xs-offset-11 {

margin-left: 91.66666667%;

}

.col-xs-offset-10 {

margin-left: 83.33333333%;

}

.col-xs-offset-9 {

margin-left: 75%;

}

.col-xs-offset-8 {

margin-left: 66.66666667%;

}

.col-xs-offset-7 {

margin-left: 58.33333333%;

}

.col-xs-offset-6 {

margin-left: 50%;

}

.col-xs-offset-5 {

margin-left: 41.66666667%;

}

.col-xs-offset-4 {

margin-left: 33.33333333%;

}

.col-xs-offset-3 {

margin-left: 25%;

}

.col-xs-offset-2 {

margin-left: 16.66666667%;

}

.col-xs-offset-1 {

margin-left: 8.33333333%;

}

.col-xs-offset-0 {

margin-left: 0;

}

@media (min-width: 768px) {

.col-sm-1, .col-sm-2, .col-sm-3, .col-sm-4, .col-sm-5, .col-sm-6, .col-sm-7, .col-sm-8, .col-sm-9, .col-sm-10, .col-sm-11, .col-sm-12 {

float: left;

}

.col-sm-12 {

width: 100%;

}

.col-sm-11 {

width: 91.66666667%;

}

.col-sm-10 {

width: 83.33333333%;

}

.col-sm-9 {

width: 75%;

}

.col-sm-8 {

width: 66.66666667%;

}

.col-sm-7 {

width: 58.33333333%;

}

.col-sm-6 {

width: 50%;

}

.col-sm-5 {

width: 41.66666667%;

}

.col-sm-4 {

width: 33.33333333%;

}

.col-sm-3 {

width: 25%;

}

.col-sm-2 {

width: 16.66666667%;

}

.col-sm-1 {

width: 8.33333333%;

}

.col-sm-pull-12 {

right: 100%;

}

.col-sm-pull-11 {

right: 91.66666667%;

}

.col-sm-pull-10 {

right: 83.33333333%;

}

.col-sm-pull-9 {

right: 75%;

}

.col-sm-pull-8 {

right: 66.66666667%;

}

.col-sm-pull-7 {

right: 58.33333333%;

}

.col-sm-pull-6 {

right: 50%;

}

.col-sm-pull-5 {

right: 41.66666667%;

}

.col-sm-pull-4 {

right: 33.33333333%;

}

.col-sm-pull-3 {

right: 25%;

}

.col-sm-pull-2 {

right: 16.66666667%;

}

.col-sm-pull-1 {

right: 8.33333333%;

}

.col-sm-pull-0 {

right: auto;

}

.col-sm-push-12 {

left: 100%;

}

.col-sm-push-11 {

left: 91.66666667%;

}

.col-sm-push-10 {

left: 83.33333333%;

}

.col-sm-push-9 {

left: 75%;

}

.col-sm-push-8 {

left: 66.66666667%;

}

.col-sm-push-7 {

left: 58.33333333%;

}

.col-sm-push-6 {

left: 50%;

}

.col-sm-push-5 {

left: 41.66666667%;

}

.col-sm-push-4 {

left: 33.33333333%;

}

.col-sm-push-3 {

left: 25%;

}

.col-sm-push-2 {

left: 16.66666667%;

}

.col-sm-push-1 {

left: 8.33333333%;

}

.col-sm-push-0 {

left: auto;

}

.col-sm-offset-12 {

margin-left: 100%;

}

.col-sm-offset-11 {

margin-left: 91.66666667%;

}

.col-sm-offset-10 {

margin-left: 83.33333333%;

}

.col-sm-offset-9 {

margin-left: 75%;

}

.col-sm-offset-8 {

margin-left: 66.66666667%;

}

.col-sm-offset-7 {

margin-left: 58.33333333%;

}

.col-sm-offset-6 {

margin-left: 50%;

}

.col-sm-offset-5 {

margin-left: 41.66666667%;

}

.col-sm-offset-4 {

margin-left: 33.33333333%;

}

.col-sm-offset-3 {

margin-left: 25%;

}

.col-sm-offset-2 {

margin-left: 16.66666667%;

}

.col-sm-offset-1 {

margin-left: 8.33333333%;

}

.col-sm-offset-0 {

margin-left: 0;

}

}

@media (min-width: 992px) {

.col-md-1, .col-md-2, .col-md-3, .col-md-4, .col-md-5, .col-md-6, .col-md-7, .col-md-8, .col-md-9, .col-md-10, .col-md-11, .col-md-12 {

float: left;

}

.col-md-12 {

width: 100%;

}

.col-md-11 {

width: 91.66666667%;

}

.col-md-10 {

width: 83.33333333%;

}

.col-md-9 {

width: 75%;

}

.col-md-8 {

width: 66.66666667%;

}

.col-md-7 {

width: 58.33333333%;

}

.col-md-6 {

width: 50%;

}

.col-md-5 {

width: 41.66666667%;

}

.col-md-4 {

width: 33.33333333%;

}

.col-md-3 {

width: 25%;

}

.col-md-2 {

width: 16.66666667%;

}

.col-md-1 {

width: 8.33333333%;

}

.col-md-pull-12 {

right: 100%;

}

.col-md-pull-11 {

right: 91.66666667%;

}

.col-md-pull-10 {

right: 83.33333333%;

}

.col-md-pull-9 {

right: 75%;

}

.col-md-pull-8 {

right: 66.66666667%;

}

.col-md-pull-7 {

right: 58.33333333%;

}

.col-md-pull-6 {

right: 50%;

}

.col-md-pull-5 {

right: 41.66666667%;

}

.col-md-pull-4 {

right: 33.33333333%;

}

.col-md-pull-3 {

right: 25%;

}

.col-md-pull-2 {

right: 16.66666667%;

}

.col-md-pull-1 {

right: 8.33333333%;

}

.col-md-pull-0 {

right: auto;

}

.col-md-push-12 {

left: 100%;

}

.col-md-push-11 {

left: 91.66666667%;

}

.col-md-push-10 {

left: 83.33333333%;

}

.col-md-push-9 {

left: 75%;

}

.col-md-push-8 {

left: 66.66666667%;

}

.col-md-push-7 {

left: 58.33333333%;

}

.col-md-push-6 {

left: 50%;

}

.col-md-push-5 {

left: 41.66666667%;

}

.col-md-push-4 {

left: 33.33333333%;

}

.col-md-push-3 {

left: 25%;

}

.col-md-push-2 {

left: 16.66666667%;

}

.col-md-push-1 {

left: 8.33333333%;

}

.col-md-push-0 {

left: auto;

}

.col-md-offset-12 {

margin-left: 100%;

}

.col-md-offset-11 {

margin-left: 91.66666667%;

}

.col-md-offset-10 {

margin-left: 83.33333333%;

}

.col-md-offset-9 {

margin-left: 75%;

}

.col-md-offset-8 {

margin-left: 66.66666667%;

}

.col-md-offset-7 {

margin-left: 58.33333333%;

}

.col-md-offset-6 {

margin-left: 50%;

}

.col-md-offset-5 {

margin-left: 41.66666667%;

}

.col-md-offset-4 {

margin-left: 33.33333333%;

}

.col-md-offset-3 {

margin-left: 25%;

}

.col-md-offset-2 {

margin-left: 16.66666667%;

}

.col-md-offset-1 {

margin-left: 8.33333333%;

}

.col-md-offset-0 {

margin-left: 0;

}

}

@media (min-width: 1200px) {

.col-lg-1, .col-lg-2, .col-lg-3, .col-lg-4, .col-lg-5, .col-lg-6, .col-lg-7, .col-lg-8, .col-lg-9, .col-lg-10, .col-lg-11, .col-lg-12 {

float: left;

}

.col-lg-12 {

width: 100%;

}

.col-lg-11 {

width: 91.66666667%;

}

.col-lg-10 {

width: 83.33333333%;

}

.col-lg-9 {

width: 75%;

}

.col-lg-8 {

width: 66.66666667%;

}

.col-lg-7 {

width: 58.33333333%;

}

.col-lg-6 {

width: 50%;

}

.col-lg-5 {

width: 41.66666667%;

}

.col-lg-4 {

width: 33.33333333%;

}

.col-lg-3 {

width: 25%;

}

.col-lg-2 {

width: 16.66666667%;

}

.col-lg-1 {

width: 8.33333333%;

}

.col-lg-pull-12 {

right: 100%;

}

.col-lg-pull-11 {

right: 91.66666667%;

}

.col-lg-pull-10 {

right: 83.33333333%;

}

.col-lg-pull-9 {

right: 75%;

}

.col-lg-pull-8 {

right: 66.66666667%;

}

.col-lg-pull-7 {

right: 58.33333333%;

}

.col-lg-pull-6 {

right: 50%;

}

.col-lg-pull-5 {

right: 41.66666667%;

}

.col-lg-pull-4 {

right: 33.33333333%;

}

.col-lg-pull-3 {

right: 25%;

}

.col-lg-pull-2 {

right: 16.66666667%;

}

.col-lg-pull-1 {

right: 8.33333333%;

}

.col-lg-pull-0 {

right: auto;

}

.col-lg-push-12 {

left: 100%;

}

.col-lg-push-11 {

left: 91.66666667%;

}

.col-lg-push-10 {

left: 83.33333333%;

}

.col-lg-push-9 {

left: 75%;

}

.col-lg-push-8 {

left: 66.66666667%;

}

.col-lg-push-7 {

left: 58.33333333%;

}

.col-lg-push-6 {

left: 50%;

}

.col-lg-push-5 {

left: 41.66666667%;

}

.col-lg-push-4 {

left: 33.33333333%;

}

.col-lg-push-3 {

left: 25%;

}

.col-lg-push-2 {

left: 16.66666667%;

}

.col-lg-push-1 {

left: 8.33333333%;

}

.col-lg-push-0 {

left: auto;

}

.col-lg-offset-12 {

margin-left: 100%;

}

.col-lg-offset-11 {

margin-left: 91.66666667%;

}

.col-lg-offset-10 {

margin-left: 83.33333333%;

}

.col-lg-offset-9 {

margin-left: 75%;

}

.col-lg-offset-8 {

margin-left: 66.66666667%;

}

.col-lg-offset-7 {

margin-left: 58.33333333%;

}

.col-lg-offset-6 {

margin-left: 50%;

}

.col-lg-offset-5 {

margin-left: 41.66666667%;

}

.col-lg-offset-4 {

margin-left: 33.33333333%;

}

.col-lg-offset-3 {

margin-left: 25%;

}

.col-lg-offset-2 {

margin-left: 16.66666667%;

}

.col-lg-offset-1 {

margin-left: 8.33333333%;

}

.col-lg-offset-0 {

margin-left: 0;

}

}

table {

background-color: transparent;

}

caption {

padding-top: 8px;

padding-bottom: 8px;

color: #777;

text-align: left;

}

th {

text-align: left;

}

.table {

width: 100%;

max-width: 100%;

margin-bottom: 20px;

}

.table > thead > tr > th,

.table > tbody > tr > th,

.table > tfoot > tr > th,

.table > thead > tr > td,

.table > tbody > tr > td,

.table > tfoot > tr > td {

padding: 8px;

line-height: 1.42857143;

vertical-align: top;

border-top: 1px solid #ddd;

}

.table > thead > tr > th {

vertical-align: bottom;

border-bottom: 2px solid #ddd;

}

.table > caption + thead > tr:first-child > th,

.table > colgroup + thead > tr:first-child > th,

.table > thead:first-child > tr:first-child > th,

.table > caption + thead > tr:first-child > td,

.table > colgroup + thead > tr:first-child > td,

.table > thead:first-child > tr:first-child > td {

border-top: 0;

}

.table > tbody + tbody {

border-top: 2px solid #ddd;

}

.table .table {

background-color: #fff;

}

.table-condensed > thead > tr > th,

.table-condensed > tbody > tr > th,

.table-condensed > tfoot > tr > th,

.table-condensed > thead > tr > td,

.table-condensed > tbody > tr > td,

.table-condensed > tfoot > tr > td {

padding: 5px;

}

.table-bordered {

border: 1px solid #ddd;

}

.table-bordered > thead > tr > th,

.table-bordered > tbody > tr > th,

.table-bordered > tfoot > tr > th,

.table-bordered > thead > tr > td,

.table-bordered > tbody > tr > td,

.table-bordered > tfoot > tr > td {

border: 1px solid #ddd;

}

.table-bordered > thead > tr > th,

.table-bordered > thead > tr > td {

border-bottom-width: 2px;

}

.table-striped > tbody > tr:nth-of-type(odd) {

background-color: #f9f9f9;

}

.table-hover > tbody > tr:hover {

background-color: #f5f5f5;

}

table col[class\*="col-"] {

position: static;

display: table-column;

float: none;

}

table td[class\*="col-"],

table th[class\*="col-"] {

position: static;

display: table-cell;

float: none;

}

.table > thead > tr > td.active,

.table > tbody > tr > td.active,

.table > tfoot > tr > td.active,

.table > thead > tr > th.active,

.table > tbody > tr > th.active,

.table > tfoot > tr > th.active,

.table > thead > tr.active > td,

.table > tbody > tr.active > td,

.table > tfoot > tr.active > td,

.table > thead > tr.active > th,

.table > tbody > tr.active > th,

.table > tfoot > tr.active > th {

background-color: #f5f5f5;

}

.table-hover > tbody > tr > td.active:hover,

.table-hover > tbody > tr > th.active:hover,

.table-hover > tbody > tr.active:hover > td,

.table-hover > tbody > tr:hover > .active,

.table-hover > tbody > tr.active:hover > th {

background-color: #e8e8e8;

}

.table > thead > tr > td.success,

.table > tbody > tr > td.success,

.table > tfoot > tr > td.success,

.table > thead > tr > th.success,

.table > tbody > tr > th.success,

.table > tfoot > tr > th.success,

.table > thead > tr.success > td,

.table > tbody > tr.success > td,

.table > tfoot > tr.success > td,

.table > thead > tr.success > th,

.table > tbody > tr.success > th,

.table > tfoot > tr.success > th {

background-color: #dff0d8;

}

.table-hover > tbody > tr > td.success:hover,

.table-hover > tbody > tr > th.success:hover,

.table-hover > tbody > tr.success:hover > td,

.table-hover > tbody > tr:hover > .success,

.table-hover > tbody > tr.success:hover > th {

background-color: #d0e9c6;

}

.table > thead > tr > td.info,

.table > tbody > tr > td.info,

.table > tfoot > tr > td.info,

.table > thead > tr > th.info,

.table > tbody > tr > th.info,

.table > tfoot > tr > th.info,

.table > thead > tr.info > td,

.table > tbody > tr.info > td,

.table > tfoot > tr.info > td,

.table > thead > tr.info > th,

.table > tbody > tr.info > th,

.table > tfoot > tr.info > th {

background-color: #d9edf7;

}

.table-hover > tbody > tr > td.info:hover,

.table-hover > tbody > tr > th.info:hover,

.table-hover > tbody > tr.info:hover > td,

.table-hover > tbody > tr:hover > .info,

.table-hover > tbody > tr.info:hover > th {

background-color: #c4e3f3;

}

.table > thead > tr > td.warning,

.table > tbody > tr > td.warning,

.table > tfoot > tr > td.warning,

.table > thead > tr > th.warning,

.table > tbody > tr > th.warning,

.table > tfoot > tr > th.warning,

.table > thead > tr.warning > td,

.table > tbody > tr.warning > td,

.table > tfoot > tr.warning > td,

.table > thead > tr.warning > th,

.table > tbody > tr.warning > th,

.table > tfoot > tr.warning > th {

background-color: #fcf8e3;

}

.table-hover > tbody > tr > td.warning:hover,

.table-hover > tbody > tr > th.warning:hover,

.table-hover > tbody > tr.warning:hover > td,

.table-hover > tbody > tr:hover > .warning,

.table-hover > tbody > tr.warning:hover > th {

background-color: #faf2cc;

}

.table > thead > tr > td.danger,

.table > tbody > tr > td.danger,

.table > tfoot > tr > td.danger,

.table > thead > tr > th.danger,

.table > tbody > tr > th.danger,

.table > tfoot > tr > th.danger,

.table > thead > tr.danger > td,

.table > tbody > tr.danger > td,

.table > tfoot > tr.danger > td,

.table > thead > tr.danger > th,

.table > tbody > tr.danger > th,

.table > tfoot > tr.danger > th {

background-color: #f2dede;

}

.table-hover > tbody > tr > td.danger:hover,

.table-hover > tbody > tr > th.danger:hover,

.table-hover > tbody > tr.danger:hover > td,

.table-hover > tbody > tr:hover > .danger,

.table-hover > tbody > tr.danger:hover > th {

background-color: #ebcccc;

}

.table-responsive {

min-height: .01%;

overflow-x: auto;

}

@media screen and (max-width: 767px) {

.table-responsive {

width: 100%;

margin-bottom: 15px;

overflow-y: hidden;

-ms-overflow-style: -ms-autohiding-scrollbar;

border: 1px solid #ddd;

}

.table-responsive > .table {

margin-bottom: 0;

}

.table-responsive > .table > thead > tr > th,

.table-responsive > .table > tbody > tr > th,

.table-responsive > .table > tfoot > tr > th,

.table-responsive > .table > thead > tr > td,

.table-responsive > .table > tbody > tr > td,

.table-responsive > .table > tfoot > tr > td {

white-space: nowrap;

}

.table-responsive > .table-bordered {

border: 0;

}

.table-responsive > .table-bordered > thead > tr > th:first-child,

.table-responsive > .table-bordered > tbody > tr > th:first-child,

.table-responsive > .table-bordered > tfoot > tr > th:first-child,

.table-responsive > .table-bordered > thead > tr > td:first-child,

.table-responsive > .table-bordered > tbody > tr > td:first-child,

.table-responsive > .table-bordered > tfoot > tr > td:first-child {

border-left: 0;

}

.table-responsive > .table-bordered > thead > tr > th:last-child,

.table-responsive > .table-bordered > tbody > tr > th:last-child,

.table-responsive > .table-bordered > tfoot > tr > th:last-child,

.table-responsive > .table-bordered > thead > tr > td:last-child,

.table-responsive > .table-bordered > tbody > tr > td:last-child,

.table-responsive > .table-bordered > tfoot > tr > td:last-child {

border-right: 0;

}

.table-responsive > .table-bordered > tbody > tr:last-child > th,

.table-responsive > .table-bordered > tfoot > tr:last-child > th,

.table-responsive > .table-bordered > tbody > tr:last-child > td,

.table-responsive > .table-bordered > tfoot > tr:last-child > td {

border-bottom: 0;

}

}

fieldset {

min-width: 0;

padding: 0;

margin: 0;

border: 0;

}

legend {

display: block;

width: 100%;

padding: 0;

margin-bottom: 20px;

font-size: 21px;

line-height: inherit;

color: #333;

border: 0;

border-bottom: 1px solid #e5e5e5;

}

label {

display: inline-block;

max-width: 100%;

margin-bottom: 5px;

font-weight: bold;

}

input[type="search"] {

-webkit-box-sizing: border-box;

-moz-box-sizing: border-box;

box-sizing: border-box;

}

input[type="radio"],

input[type="checkbox"] {

margin: 4px 0 0;

margin-top: 1px \9;

line-height: normal;

}

input[type="file"] {

display: block;

}

input[type="range"] {

display: block;

width: 100%;

}

select[multiple],

select[size] {

height: auto;

}

input[type="file"]:focus,

input[type="radio"]:focus,

input[type="checkbox"]:focus {

outline: 5px auto -webkit-focus-ring-color;

outline-offset: -2px;

}

output {

display: block;

padding-top: 7px;

font-size: 14px;

line-height: 1.42857143;

color: #555;

}

.form-control {

display: block;

width: 100%;

height: 34px;

padding: 6px 12px;

font-size: 14px;

line-height: 1.42857143;

color: #555;

background-color: #fff;

background-image: none;

border: 1px solid #ccc;

border-radius: 4px;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

-webkit-transition: border-color ease-in-out .15s, -webkit-box-shadow ease-in-out .15s;

-o-transition: border-color ease-in-out .15s, box-shadow ease-in-out .15s;

transition: border-color ease-in-out .15s, box-shadow ease-in-out .15s;

}

.form-control:focus {

border-color: #66afe9;

outline: 0;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 8px rgba(102, 175, 233, .6);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 8px rgba(102, 175, 233, .6);

}

.form-control::-moz-placeholder {

color: #999;

opacity: 1;

}

.form-control:-ms-input-placeholder {

color: #999;

}

.form-control::-webkit-input-placeholder {

color: #999;

}

.form-control::-ms-expand {

background-color: transparent;

border: 0;

}

.form-control[disabled],

.form-control[readonly],

fieldset[disabled] .form-control {

background-color: #eee;

opacity: 1;

}

.form-control[disabled],

fieldset[disabled] .form-control {

cursor: not-allowed;

}

textarea.form-control {

height: auto;

}

input[type="search"] {

-webkit-appearance: none;

}

@media screen and (-webkit-min-device-pixel-ratio: 0) {

input[type="date"].form-control,

input[type="time"].form-control,

input[type="datetime-local"].form-control,

input[type="month"].form-control {

line-height: 34px;

}

input[type="date"].input-sm,

input[type="time"].input-sm,

input[type="datetime-local"].input-sm,

input[type="month"].input-sm,

.input-group-sm input[type="date"],

.input-group-sm input[type="time"],

.input-group-sm input[type="datetime-local"],

.input-group-sm input[type="month"] {

line-height: 30px;

}

input[type="date"].input-lg,

input[type="time"].input-lg,

input[type="datetime-local"].input-lg,

input[type="month"].input-lg,

.input-group-lg input[type="date"],

.input-group-lg input[type="time"],

.input-group-lg input[type="datetime-local"],

.input-group-lg input[type="month"] {

line-height: 46px;

}

}

.form-group {

margin-bottom: 15px;

}

.radio,

.checkbox {

position: relative;

display: block;

margin-top: 10px;

margin-bottom: 10px;

}

.radio label,

.checkbox label {

min-height: 20px;

padding-left: 20px;

margin-bottom: 0;

font-weight: normal;

cursor: pointer;

}

.radio input[type="radio"],

.radio-inline input[type="radio"],

.checkbox input[type="checkbox"],

.checkbox-inline input[type="checkbox"] {

position: absolute;

margin-top: 4px \9;

margin-left: -20px;

}

.radio + .radio,

.checkbox + .checkbox {

margin-top: -5px;

}

.radio-inline,

.checkbox-inline {

position: relative;

display: inline-block;

padding-left: 20px;

margin-bottom: 0;

font-weight: normal;

vertical-align: middle;

cursor: pointer;

}

.radio-inline + .radio-inline,

.checkbox-inline + .checkbox-inline {

margin-top: 0;

margin-left: 10px;

}

input[type="radio"][disabled],

input[type="checkbox"][disabled],

input[type="radio"].disabled,

input[type="checkbox"].disabled,

fieldset[disabled] input[type="radio"],

fieldset[disabled] input[type="checkbox"] {

cursor: not-allowed;

}

.radio-inline.disabled,

.checkbox-inline.disabled,

fieldset[disabled] .radio-inline,

fieldset[disabled] .checkbox-inline {

cursor: not-allowed;

}

.radio.disabled label,

.checkbox.disabled label,

fieldset[disabled] .radio label,

fieldset[disabled] .checkbox label {

cursor: not-allowed;

}

.form-control-static {

min-height: 34px;

padding-top: 7px;

padding-bottom: 7px;

margin-bottom: 0;

}

.form-control-static.input-lg,

.form-control-static.input-sm {

padding-right: 0;

padding-left: 0;

}

.input-sm {

height: 30px;

padding: 5px 10px;

font-size: 12px;

line-height: 1.5;

border-radius: 3px;

}

select.input-sm {

height: 30px;

line-height: 30px;

}

textarea.input-sm,

select[multiple].input-sm {

height: auto;

}

.form-group-sm .form-control {

height: 30px;

padding: 5px 10px;

font-size: 12px;

line-height: 1.5;

border-radius: 3px;

}

.form-group-sm select.form-control {

height: 30px;

line-height: 30px;

}

.form-group-sm textarea.form-control,

.form-group-sm select[multiple].form-control {

height: auto;

}

.form-group-sm .form-control-static {

height: 30px;

min-height: 32px;

padding: 6px 10px;

font-size: 12px;

line-height: 1.5;

}

.input-lg {

height: 46px;

padding: 10px 16px;

font-size: 18px;

line-height: 1.3333333;

border-radius: 6px;

}

select.input-lg {

height: 46px;

line-height: 46px;

}

textarea.input-lg,

select[multiple].input-lg {

height: auto;

}

.form-group-lg .form-control {

height: 46px;

padding: 10px 16px;

font-size: 18px;

line-height: 1.3333333;

border-radius: 6px;

}

.form-group-lg select.form-control {

height: 46px;

line-height: 46px;

}

.form-group-lg textarea.form-control,

.form-group-lg select[multiple].form-control {

height: auto;

}

.form-group-lg .form-control-static {

height: 46px;

min-height: 38px;

padding: 11px 16px;

font-size: 18px;

line-height: 1.3333333;

}

.has-feedback {

position: relative;

}

.has-feedback .form-control {

padding-right: 42.5px;

}

.form-control-feedback {

position: absolute;

top: 0;

right: 0;

z-index: 2;

display: block;

width: 34px;

height: 34px;

line-height: 34px;

text-align: center;

pointer-events: none;

}

.input-lg + .form-control-feedback,

.input-group-lg + .form-control-feedback,

.form-group-lg .form-control + .form-control-feedback {

width: 46px;

height: 46px;

line-height: 46px;

}

.input-sm + .form-control-feedback,

.input-group-sm + .form-control-feedback,

.form-group-sm .form-control + .form-control-feedback {

width: 30px;

height: 30px;

line-height: 30px;

}

.has-success .help-block,

.has-success .control-label,

.has-success .radio,

.has-success .checkbox,

.has-success .radio-inline,

.has-success .checkbox-inline,

.has-success.radio label,

.has-success.checkbox label,

.has-success.radio-inline label,

.has-success.checkbox-inline label {

color: #3c763d;

}

.has-success .form-control {

border-color: #3c763d;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

}

.has-success .form-control:focus {

border-color: #2b542c;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #67b168;

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #67b168;

}

.has-success .input-group-addon {

color: #3c763d;

background-color: #dff0d8;

border-color: #3c763d;

}

.has-success .form-control-feedback {

color: #3c763d;

}

.has-warning .help-block,

.has-warning .control-label,

.has-warning .radio,

.has-warning .checkbox,

.has-warning .radio-inline,

.has-warning .checkbox-inline,

.has-warning.radio label,

.has-warning.checkbox label,

.has-warning.radio-inline label,

.has-warning.checkbox-inline label {

color: #8a6d3b;

}

.has-warning .form-control {

border-color: #8a6d3b;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

}

.has-warning .form-control:focus {

border-color: #66512c;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #c0a16b;

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #c0a16b;

}

.has-warning .input-group-addon {

color: #8a6d3b;

background-color: #fcf8e3;

border-color: #8a6d3b;

}

.has-warning .form-control-feedback {

color: #8a6d3b;

}

.has-error .help-block,

.has-error .control-label,

.has-error .radio,

.has-error .checkbox,

.has-error .radio-inline,

.has-error .checkbox-inline,

.has-error.radio label,

.has-error.checkbox label,

.has-error.radio-inline label,

.has-error.checkbox-inline label {

color: #a94442;

}

.has-error .form-control {

border-color: #a94442;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075);

}

.has-error .form-control:focus {

border-color: #843534;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #ce8483;

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .075), 0 0 6px #ce8483;

}

.has-error .input-group-addon {

color: #a94442;

background-color: #f2dede;

border-color: #a94442;

}

.has-error .form-control-feedback {

color: #a94442;

}

.has-feedback label ~ .form-control-feedback {

top: 25px;

}

.has-feedback label.sr-only ~ .form-control-feedback {

top: 0;

}

.help-block {

display: block;

margin-top: 5px;

margin-bottom: 10px;

color: #737373;

}

@media (min-width: 768px) {

.form-inline .form-group {

display: inline-block;

margin-bottom: 0;

vertical-align: middle;

}

.form-inline .form-control {

display: inline-block;

width: auto;

vertical-align: middle;

}

.form-inline .form-control-static {

display: inline-block;

}

.form-inline .input-group {

display: inline-table;

vertical-align: middle;

}

.form-inline .input-group .input-group-addon,

.form-inline .input-group .input-group-btn,

.form-inline .input-group .form-control {

width: auto;

}

.form-inline .input-group > .form-control {

width: 100%;

}

.form-inline .control-label {

margin-bottom: 0;

vertical-align: middle;

}

.form-inline .radio,

.form-inline .checkbox {

display: inline-block;

margin-top: 0;

margin-bottom: 0;

vertical-align: middle;

}

.form-inline .radio label,

.form-inline .checkbox label {

padding-left: 0;

}

.form-inline .radio input[type="radio"],

.form-inline .checkbox input[type="checkbox"] {

position: relative;

margin-left: 0;

}

.form-inline .has-feedback .form-control-feedback {

top: 0;

}

}

.form-horizontal .radio,

.form-horizontal .checkbox,

.form-horizontal .radio-inline,

.form-horizontal .checkbox-inline {

padding-top: 7px;

margin-top: 0;

margin-bottom: 0;

}

.form-horizontal .radio,

.form-horizontal .checkbox {

min-height: 27px;

}

.form-horizontal .form-group {

margin-right: -15px;

margin-left: -15px;

}

@media (min-width: 768px) {

.form-horizontal .control-label {

padding-top: 7px;

margin-bottom: 0;

text-align: right;

}

}

.form-horizontal .has-feedback .form-control-feedback {

right: 15px;

}

@media (min-width: 768px) {

.form-horizontal .form-group-lg .control-label {

padding-top: 11px;

font-size: 18px;

}

}

@media (min-width: 768px) {

.form-horizontal .form-group-sm .control-label {

padding-top: 6px;

font-size: 12px;

}

}

.btn {

display: inline-block;

padding: 6px 12px;

margin-bottom: 0;

font-size: 14px;

font-weight: normal;

line-height: 1.42857143;

text-align: center;

white-space: nowrap;

vertical-align: middle;

-ms-touch-action: manipulation;

touch-action: manipulation;

cursor: pointer;

-webkit-user-select: none;

-moz-user-select: none;

-ms-user-select: none;

user-select: none;

background-image: none;

border: 1px solid transparent;

border-radius: 4px;

}

.btn:focus,

.btn:active:focus,

.btn.active:focus,

.btn.focus,

.btn:active.focus,

.btn.active.focus {

outline: 5px auto -webkit-focus-ring-color;

outline-offset: -2px;

}

.btn:hover,

.btn:focus,

.btn.focus {

color: #333;

text-decoration: none;

}

.btn:active,

.btn.active {

background-image: none;

outline: 0;

-webkit-box-shadow: inset 0 3px 5px rgba(0, 0, 0, .125);

box-shadow: inset 0 3px 5px rgba(0, 0, 0, .125);

}

.btn.disabled,

.btn[disabled],

fieldset[disabled] .btn {

cursor: not-allowed;

filter: alpha(opacity=65);

-webkit-box-shadow: none;

box-shadow: none;

opacity: .65;

}

a.btn.disabled,

fieldset[disabled] a.btn {

pointer-events: none;

}

.btn-default {

color: #333;

background-color: #fff;

border-color: #ccc;

}

.btn-default:focus,

.btn-default.focus {

color: #333;

background-color: #e6e6e6;

border-color: #8c8c8c;

}

.btn-default:hover {

color: #333;

background-color: #e6e6e6;

border-color: #adadad;

}

.btn-default:active,

.btn-default.active,

.open > .dropdown-toggle.btn-default {

color: #333;

background-color: #e6e6e6;

border-color: #adadad;

}

.btn-default:active:hover,

.btn-default.active:hover,

.open > .dropdown-toggle.btn-default:hover,

.btn-default:active:focus,

.btn-default.active:focus,

.open > .dropdown-toggle.btn-default:focus,

.btn-default:active.focus,

.btn-default.active.focus,

.open > .dropdown-toggle.btn-default.focus {

color: #333;

background-color: #d4d4d4;

border-color: #8c8c8c;

}

.btn-default:active,

.btn-default.active,

.open > .dropdown-toggle.btn-default {

background-image: none;

}

.btn-default.disabled:hover,

.btn-default[disabled]:hover,

fieldset[disabled] .btn-default:hover,

.btn-default.disabled:focus,

.btn-default[disabled]:focus,

fieldset[disabled] .btn-default:focus,

.btn-default.disabled.focus,

.btn-default[disabled].focus,

fieldset[disabled] .btn-default.focus {

background-color: #fff;

border-color: #ccc;

}

.btn-default .badge {

color: #fff;

background-color: #333;

}

.btn-primary {

color: #fff;

background-color: #337ab7;

border-color: #2e6da4;

}

.btn-primary:focus,

.btn-primary.focus {

color: #fff;

background-color: #286090;

border-color: #122b40;

}

.btn-primary:hover {

color: #fff;

background-color: #286090;

border-color: #204d74;

}

.btn-primary:active,

.btn-primary.active,

.open > .dropdown-toggle.btn-primary {

color: #fff;

background-color: #286090;

border-color: #204d74;

}

.btn-primary:active:hover,

.btn-primary.active:hover,

.open > .dropdown-toggle.btn-primary:hover,

.btn-primary:active:focus,

.btn-primary.active:focus,

.open > .dropdown-toggle.btn-primary:focus,

.btn-primary:active.focus,

.btn-primary.active.focus,

.open > .dropdown-toggle.btn-primary.focus {

color: #fff;

background-color: #204d74;

border-color: #122b40;

}

.btn-primary:active,

.btn-primary.active,

.open > .dropdown-toggle.btn-primary {

background-image: none;

}

.btn-primary.disabled:hover,

.btn-primary[disabled]:hover,

fieldset[disabled] .btn-primary:hover,

.btn-primary.disabled:focus,

.btn-primary[disabled]:focus,

fieldset[disabled] .btn-primary:focus,

.btn-primary.disabled.focus,

.btn-primary[disabled].focus,

fieldset[disabled] .btn-primary.focus {

background-color: #337ab7;

border-color: #2e6da4;

}

.btn-primary .badge {

color: #337ab7;

background-color: #fff;

}

.btn-success {

color: #fff;

background-color: #5cb85c;

border-color: #4cae4c;

}

.btn-success:focus,

.btn-success.focus {

color: #fff;

background-color: #449d44;

border-color: #255625;

}

.btn-success:hover {

color: #fff;

background-color: #449d44;

border-color: #398439;

}

.btn-success:active,

.btn-success.active,

.open > .dropdown-toggle.btn-success {

color: #fff;

background-color: #449d44;

border-color: #398439;

}

.btn-success:active:hover,

.btn-success.active:hover,

.open > .dropdown-toggle.btn-success:hover,

.btn-success:active:focus,

.btn-success.active:focus,

.open > .dropdown-toggle.btn-success:focus,

.btn-success:active.focus,

.btn-success.active.focus,

.open > .dropdown-toggle.btn-success.focus {

color: #fff;

background-color: #398439;

border-color: #255625;

}

.btn-success:active,

.btn-success.active,

.open > .dropdown-toggle.btn-success {

background-image: none;

}

.btn-success.disabled:hover,

.btn-success[disabled]:hover,

fieldset[disabled] .btn-success:hover,

.btn-success.disabled:focus,

.btn-success[disabled]:focus,

fieldset[disabled] .btn-success:focus,

.btn-success.disabled.focus,

.btn-success[disabled].focus,

fieldset[disabled] .btn-success.focus {

background-color: #5cb85c;

border-color: #4cae4c;

}

.btn-success .badge {

color: #5cb85c;

background-color: #fff;

}

.btn-info {

color: #fff;

background-color: #5bc0de;

border-color: #46b8da;

}

.btn-info:focus,

.btn-info.focus {

color: #fff;

background-color: #31b0d5;

border-color: #1b6d85;

}

.btn-info:hover {

color: #fff;

background-color: #31b0d5;

border-color: #269abc;

}

.btn-info:active,

.btn-info.active,

.open > .dropdown-toggle.btn-info {

color: #fff;

background-color: #31b0d5;

border-color: #269abc;

}

.btn-info:active:hover,

.btn-info.active:hover,

.open > .dropdown-toggle.btn-info:hover,

.btn-info:active:focus,

.btn-info.active:focus,

.open > .dropdown-toggle.btn-info:focus,

.btn-info:active.focus,

.btn-info.active.focus,

.open > .dropdown-toggle.btn-info.focus {

color: #fff;

background-color: #269abc;

border-color: #1b6d85;

}

.btn-info:active,

.btn-info.active,

.open > .dropdown-toggle.btn-info {

background-image: none;

}

.btn-info.disabled:hover,

.btn-info[disabled]:hover,

fieldset[disabled] .btn-info:hover,

.btn-info.disabled:focus,

.btn-info[disabled]:focus,

fieldset[disabled] .btn-info:focus,

.btn-info.disabled.focus,

.btn-info[disabled].focus,

fieldset[disabled] .btn-info.focus {

background-color: #5bc0de;

border-color: #46b8da;

}

.btn-info .badge {

color: #5bc0de;

background-color: #fff;

}

.btn-warning {

color: #fff;

background-color: #f0ad4e;

border-color: #eea236;

}

.btn-warning:focus,

.btn-warning.focus {

color: #fff;

background-color: #ec971f;

border-color: #985f0d;

}

.btn-warning:hover {

color: #fff;

background-color: #ec971f;

border-color: #d58512;

}

.btn-warning:active,

.btn-warning.active,

.open > .dropdown-toggle.btn-warning {

color: #fff;

background-color: #ec971f;

border-color: #d58512;

}

.btn-warning:active:hover,

.btn-warning.active:hover,

.open > .dropdown-toggle.btn-warning:hover,

.btn-warning:active:focus,

.btn-warning.active:focus,

.open > .dropdown-toggle.btn-warning:focus,

.btn-warning:active.focus,

.btn-warning.active.focus,

.open > .dropdown-toggle.btn-warning.focus {

color: #fff;

background-color: #d58512;

border-color: #985f0d;

}

.btn-warning:active,

.btn-warning.active,

.open > .dropdown-toggle.btn-warning {

background-image: none;

}

.btn-warning.disabled:hover,

.btn-warning[disabled]:hover,

fieldset[disabled] .btn-warning:hover,

.btn-warning.disabled:focus,

.btn-warning[disabled]:focus,

fieldset[disabled] .btn-warning:focus,

.btn-warning.disabled.focus,

.btn-warning[disabled].focus,

fieldset[disabled] .btn-warning.focus {

background-color: #f0ad4e;

border-color: #eea236;

}

.btn-warning .badge {

color: #f0ad4e;

background-color: #fff;

}

.btn-danger {

color: #fff;

background-color: #d9534f;

border-color: #d43f3a;

}

.btn-danger:focus,

.btn-danger.focus {

color: #fff;

background-color: #c9302c;

border-color: #761c19;

}

.btn-danger:hover {

color: #fff;

background-color: #c9302c;

border-color: #ac2925;

}

.btn-danger:active,

.btn-danger.active,

.open > .dropdown-toggle.btn-danger {

color: #fff;

background-color: #c9302c;

border-color: #ac2925;

}

.btn-danger:active:hover,

.btn-danger.active:hover,

.open > .dropdown-toggle.btn-danger:hover,

.btn-danger:active:focus,

.btn-danger.active:focus,

.open > .dropdown-toggle.btn-danger:focus,

.btn-danger:active.focus,

.btn-danger.active.focus,

.open > .dropdown-toggle.btn-danger.focus {

color: #fff;

background-color: #ac2925;

border-color: #761c19;

}

.btn-danger:active,

.btn-danger.active,

.open > .dropdown-toggle.btn-danger {

background-image: none;

}

.btn-danger.disabled:hover,

.btn-danger[disabled]:hover,

fieldset[disabled] .btn-danger:hover,

.btn-danger.disabled:focus,

.btn-danger[disabled]:focus,

fieldset[disabled] .btn-danger:focus,

.btn-danger.disabled.focus,

.btn-danger[disabled].focus,

fieldset[disabled] .btn-danger.focus {

background-color: #d9534f;

border-color: #d43f3a;

}

.btn-danger .badge {

color: #d9534f;

background-color: #fff;

}

.btn-link {

font-weight: normal;

color: #337ab7;

border-radius: 0;

}

.btn-link,

.btn-link:active,

.btn-link.active,

.btn-link[disabled],

fieldset[disabled] .btn-link {

background-color: transparent;

-webkit-box-shadow: none;

box-shadow: none;

}

.btn-link,

.btn-link:hover,

.btn-link:focus,

.btn-link:active {

border-color: transparent;

}

.btn-link:hover,

.btn-link:focus {

color: #23527c;

text-decoration: underline;

background-color: transparent;

}

.btn-link[disabled]:hover,

fieldset[disabled] .btn-link:hover,

.btn-link[disabled]:focus,

fieldset[disabled] .btn-link:focus {

color: #777;

text-decoration: none;

}

.btn-lg,

.btn-group-lg > .btn {

padding: 10px 16px;

font-size: 18px;

line-height: 1.3333333;

border-radius: 6px;

}

.btn-sm,

.btn-group-sm > .btn {

padding: 5px 10px;

font-size: 12px;

line-height: 1.5;

border-radius: 3px;

}

.btn-xs,

.btn-group-xs > .btn {

padding: 1px 5px;

font-size: 12px;

line-height: 1.5;

border-radius: 3px;

}

.btn-block {

display: block;

width: 100%;

}

.btn-block + .btn-block {

margin-top: 5px;

}

input[type="submit"].btn-block,

input[type="reset"].btn-block,

input[type="button"].btn-block {

width: 100%;

}

.fade {

opacity: 0;

-webkit-transition: opacity .15s linear;

-o-transition: opacity .15s linear;

transition: opacity .15s linear;

}

.fade.in {

opacity: 1;

}

.collapse {

display: none;

}

.collapse.in {

display: block;

}

tr.collapse.in {

display: table-row;

}

tbody.collapse.in {

display: table-row-group;

}

.collapsing {

position: relative;

height: 0;

overflow: hidden;

-webkit-transition-timing-function: ease;

-o-transition-timing-function: ease;

transition-timing-function: ease;

-webkit-transition-duration: .35s;

-o-transition-duration: .35s;

transition-duration: .35s;

-webkit-transition-property: height, visibility;

-o-transition-property: height, visibility;

transition-property: height, visibility;

}

.caret {

display: inline-block;

width: 0;

height: 0;

margin-left: 2px;

vertical-align: middle;

border-top: 4px dashed;

border-top: 4px solid \9;

border-right: 4px solid transparent;

border-left: 4px solid transparent;

}

.dropup,

.dropdown {

position: relative;

}

.dropdown-toggle:focus {

outline: 0;

}

.dropdown-menu {

position: absolute;

top: 100%;

left: 0;

z-index: 1000;

display: none;

float: left;

min-width: 160px;

padding: 5px 0;

margin: 2px 0 0;

font-size: 14px;

text-align: left;

list-style: none;

background-color: #fff;

-webkit-background-clip: padding-box;

background-clip: padding-box;

border: 1px solid #ccc;

border: 1px solid rgba(0, 0, 0, .15);

border-radius: 4px;

-webkit-box-shadow: 0 6px 12px rgba(0, 0, 0, .175);

box-shadow: 0 6px 12px rgba(0, 0, 0, .175);

}

.dropdown-menu.pull-right {

right: 0;

left: auto;

}

.dropdown-menu .divider {

height: 1px;

margin: 9px 0;

overflow: hidden;

background-color: #e5e5e5;

}

.dropdown-menu > li > a {

display: block;

padding: 3px 20px;

clear: both;

font-weight: normal;

line-height: 1.42857143;

color: #333;

white-space: nowrap;

}

.dropdown-menu > li > a:hover,

.dropdown-menu > li > a:focus {

color: #262626;

text-decoration: none;

background-color: #f5f5f5;

}

.dropdown-menu > .active > a,

.dropdown-menu > .active > a:hover,

.dropdown-menu > .active > a:focus {

color: #fff;

text-decoration: none;

background-color: #337ab7;

outline: 0;

}

.dropdown-menu > .disabled > a,

.dropdown-menu > .disabled > a:hover,

.dropdown-menu > .disabled > a:focus {

color: #777;

}

.dropdown-menu > .disabled > a:hover,

.dropdown-menu > .disabled > a:focus {

text-decoration: none;

cursor: not-allowed;

background-color: transparent;

background-image: none;

filter: progid:DXImageTransform.Microsoft.gradient(enabled=false);

}

.open > .dropdown-menu {

display: block;

}

.open > a {

outline: 0;

}

.dropdown-menu-right {

right: 0;

left: auto;

}

.dropdown-menu-left {

right: auto;

left: 0;

}

.dropdown-header {

display: block;

padding: 3px 20px;

font-size: 12px;

line-height: 1.42857143;

color: #777;

white-space: nowrap;

}

.dropdown-backdrop {

position: fixed;

top: 0;

right: 0;

bottom: 0;

left: 0;

z-index: 990;

}

.pull-right > .dropdown-menu {

right: 0;

left: auto;

}

.dropup .caret,

.navbar-fixed-bottom .dropdown .caret {

content: "";

border-top: 0;

border-bottom: 4px dashed;

border-bottom: 4px solid \9;

}

.dropup .dropdown-menu,

.navbar-fixed-bottom .dropdown .dropdown-menu {

top: auto;

bottom: 100%;

margin-bottom: 2px;

}

@media (min-width: 768px) {

.navbar-right .dropdown-menu {

right: 0;

left: auto;

}

.navbar-right .dropdown-menu-left {

right: auto;

left: 0;

}

}

.btn-group,

.btn-group-vertical {

position: relative;

display: inline-block;

vertical-align: middle;

}

.btn-group > .btn,

.btn-group-vertical > .btn {

position: relative;

float: left;

}

.btn-group > .btn:hover,

.btn-group-vertical > .btn:hover,

.btn-group > .btn:focus,

.btn-group-vertical > .btn:focus,

.btn-group > .btn:active,

.btn-group-vertical > .btn:active,

.btn-group > .btn.active,

.btn-group-vertical > .btn.active {

z-index: 2;

}

.btn-group .btn + .btn,

.btn-group .btn + .btn-group,

.btn-group .btn-group + .btn,

.btn-group .btn-group + .btn-group {

margin-left: -1px;

}

.btn-toolbar {

margin-left: -5px;

}

.btn-toolbar .btn,

.btn-toolbar .btn-group,

.btn-toolbar .input-group {

float: left;

}

.btn-toolbar > .btn,

.btn-toolbar > .btn-group,

.btn-toolbar > .input-group {

margin-left: 5px;

}

.btn-group > .btn:not(:first-child):not(:last-child):not(.dropdown-toggle) {

border-radius: 0;

}

.btn-group > .btn:first-child {

margin-left: 0;

}

.btn-group > .btn:first-child:not(:last-child):not(.dropdown-toggle) {

border-top-right-radius: 0;

border-bottom-right-radius: 0;

}

.btn-group > .btn:last-child:not(:first-child),

.btn-group > .dropdown-toggle:not(:first-child) {

border-top-left-radius: 0;

border-bottom-left-radius: 0;

}

.btn-group > .btn-group {

float: left;

}

.btn-group > .btn-group:not(:first-child):not(:last-child) > .btn {

border-radius: 0;

}

.btn-group > .btn-group:first-child:not(:last-child) > .btn:last-child,

.btn-group > .btn-group:first-child:not(:last-child) > .dropdown-toggle {

border-top-right-radius: 0;

border-bottom-right-radius: 0;

}

.btn-group > .btn-group:last-child:not(:first-child) > .btn:first-child {

border-top-left-radius: 0;

border-bottom-left-radius: 0;

}

.btn-group .dropdown-toggle:active,

.btn-group.open .dropdown-toggle {

outline: 0;

}

.btn-group > .btn + .dropdown-toggle {

padding-right: 8px;

padding-left: 8px;

}

.btn-group > .btn-lg + .dropdown-toggle {

padding-right: 12px;

padding-left: 12px;

}

.btn-group.open .dropdown-toggle {

-webkit-box-shadow: inset 0 3px 5px rgba(0, 0, 0, .125);

box-shadow: inset 0 3px 5px rgba(0, 0, 0, .125);

}

.btn-group.open .dropdown-toggle.btn-link {

-webkit-box-shadow: none;

box-shadow: none;

}

.btn .caret {

margin-left: 0;

}

.btn-lg .caret {

border-width: 5px 5px 0;

border-bottom-width: 0;

}

.dropup .btn-lg .caret {

border-width: 0 5px 5px;

}

.btn-group-vertical > .btn,

.btn-group-vertical > .btn-group,

.btn-group-vertical > .btn-group > .btn {

display: block;

float: none;

width: 100%;

max-width: 100%;

}

.btn-group-vertical > .btn-group > .btn {

float: none;

}

.btn-group-vertical > .btn + .btn,

.btn-group-vertical > .btn + .btn-group,

.btn-group-vertical > .btn-group + .btn,

.btn-group-vertical > .btn-group + .btn-group {

margin-top: -1px;

margin-left: 0;

}

.btn-group-vertical > .btn:not(:first-child):not(:last-child) {

border-radius: 0;

}

.btn-group-vertical > .btn:first-child:not(:last-child) {

border-top-left-radius: 4px;

border-top-right-radius: 4px;

border-bottom-right-radius: 0;

border-bottom-left-radius: 0;

}

.btn-group-vertical > .btn:last-child:not(:first-child) {

border-top-left-radius: 0;

border-top-right-radius: 0;

border-bottom-right-radius: 4px;

border-bottom-left-radius: 4px;

}

.btn-group-vertical > .btn-group:not(:first-child):not(:last-child) > .btn {

border-radius: 0;

}

.btn-group-vertical > .btn-group:first-child:not(:last-child) > .btn:last-child,

.btn-group-vertical > .btn-group:first-child:not(:last-child) > .dropdown-toggle {

border-bottom-right-radius: 0;

border-bottom-left-radius: 0;

}

.btn-group-vertical > .btn-group:last-child:not(:first-child) > .btn:first-child {

border-top-left-radius: 0;

border-top-right-radius: 0;

}

.btn-group-justified {

display: table;

width: 100%;

table-layout: fixed;

border-collapse: separate;

}

.btn-group-justified > .btn,

.btn-group-justified > .btn-group {

display: table-cell;

float: none;

width: 1%;

}

.btn-group-justified > .btn-group .btn {

width: 100%;

}

.btn-group-justified > .btn-group .dropdown-menu {

left: auto;

}

[data-toggle="buttons"] > .btn input[type="radio"],

[data-toggle="buttons"] > .btn-group > .btn input[type="radio"],

[data-toggle="buttons"] > .btn input[type="checkbox"],

[data-toggle="buttons"] > .btn-group > .btn input[type="checkbox"] {

position: absolute;

clip: rect(0, 0, 0, 0);

pointer-events: none;

}

.input-group {

position: relative;

display: table;

border-collapse: separate;

}

.input-group[class\*="col-"] {

float: none;

padding-right: 0;

padding-left: 0;

}

.input-group .form-control {

position: relative;

z-index: 2;

float: left;

width: 100%;

margin-bottom: 0;

}

.input-group .form-control:focus {

z-index: 3;

}

.input-group-lg > .form-control,

.input-group-lg > .input-group-addon,

.input-group-lg > .input-group-btn > .btn {

height: 46px;

padding: 10px 16px;

font-size: 18px;

line-height: 1.3333333;

border-radius: 6px;

}

select.input-group-lg > .form-control,

select.input-group-lg > .input-group-addon,

select.input-group-lg > .input-group-btn > .btn {

height: 46px;

line-height: 46px;

}

textarea.input-group-lg > .form-control,

textarea.input-group-lg > .input-group-addon,

textarea.input-group-lg > .input-group-btn > .btn,

select[multiple].input-group-lg > .form-control,

select[multiple].input-group-lg > .input-group-addon,

select[multiple].input-group-lg > .input-group-btn > .btn {

height: auto;

}

.input-group-sm > .form-control,

.input-group-sm > .input-group-addon,

.input-group-sm > .input-group-btn > .btn {

height: 30px;

padding: 5px 10px;

font-size: 12px;

line-height: 1.5;

border-radius: 3px;

}

select.input-group-sm > .form-control,

select.input-group-sm > .input-group-addon,

select.input-group-sm > .input-group-btn > .btn {

height: 30px;

line-height: 30px;

}

textarea.input-group-sm > .form-control,

textarea.input-group-sm > .input-group-addon,

textarea.input-group-sm > .input-group-btn > .btn,

select[multiple].input-group-sm > .form-control,

select[multiple].input-group-sm > .input-group-addon,

select[multiple].input-group-sm > .input-group-btn > .btn {

height: auto;

}

.input-group-addon,

.input-group-btn,

.input-group .form-control {

display: table-cell;

}

.input-group-addon:not(:first-child):not(:last-child),

.input-group-btn:not(:first-child):not(:last-child),

.input-group .form-control:not(:first-child):not(:last-child) {

border-radius: 0;

}

.input-group-addon,

.input-group-btn {

width: 1%;

white-space: nowrap;

vertical-align: middle;

}

.input-group-addon {

padding: 6px 12px;

font-size: 14px;

font-weight: normal;

line-height: 1;

color: #555;

text-align: center;

background-color: #eee;

border: 1px solid #ccc;

border-radius: 4px;

}

.input-group-addon.input-sm {

padding: 5px 10px;

font-size: 12px;

border-radius: 3px;

}

.input-group-addon.input-lg {

padding: 10px 16px;

font-size: 18px;

border-radius: 6px;

}

.input-group-addon input[type="radio"],

.input-group-addon input[type="checkbox"] {

margin-top: 0;

}

.input-group .form-control:first-child,

.input-group-addon:first-child,

.input-group-btn:first-child > .btn,

.input-group-btn:first-child > .btn-group > .btn,

.input-group-btn:first-child > .dropdown-toggle,

.input-group-btn:last-child > .btn:not(:last-child):not(.dropdown-toggle),

.input-group-btn:last-child > .btn-group:not(:last-child) > .btn {

border-top-right-radius: 0;

border-bottom-right-radius: 0;

}

.input-group-addon:first-child {

border-right: 0;

}

.input-group .form-control:last-child,

.input-group-addon:last-child,

.input-group-btn:last-child > .btn,

.input-group-btn:last-child > .btn-group > .btn,

.input-group-btn:last-child > .dropdown-toggle,

.input-group-btn:first-child > .btn:not(:first-child),

.input-group-btn:first-child > .btn-group:not(:first-child) > .btn {

border-top-left-radius: 0;

border-bottom-left-radius: 0;

}

.input-group-addon:last-child {

border-left: 0;

}

.input-group-btn {

position: relative;

font-size: 0;

white-space: nowrap;

}

.input-group-btn > .btn {

position: relative;

}

.input-group-btn > .btn + .btn {

margin-left: -1px;

}

.input-group-btn > .btn:hover,

.input-group-btn > .btn:focus,

.input-group-btn > .btn:active {

z-index: 2;

}

.input-group-btn:first-child > .btn,

.input-group-btn:first-child > .btn-group {

margin-right: -1px;

}

.input-group-btn:last-child > .btn,

.input-group-btn:last-child > .btn-group {

z-index: 2;

margin-left: -1px;

}

.nav {

padding-left: 0;

margin-bottom: 0;

list-style: none;

}

.nav > li {

position: relative;

display: block;

}

.nav > li > a {

position: relative;

display: block;

padding: 10px 15px;

}

.nav > li > a:hover,

.nav > li > a:focus {

text-decoration: none;

background-color: #eee;

}

.nav > li.disabled > a {

color: #777;

}

.nav > li.disabled > a:hover,

.nav > li.disabled > a:focus {

color: #777;

text-decoration: none;

cursor: not-allowed;

background-color: transparent;

}

.nav .open > a,

.nav .open > a:hover,

.nav .open > a:focus {

background-color: #eee;

border-color: #337ab7;

}

.nav .nav-divider {

height: 1px;

margin: 9px 0;

overflow: hidden;

background-color: #e5e5e5;

}

.nav > li > a > img {

max-width: none;

}

.nav-tabs {

border-bottom: 1px solid #ddd;

}

.nav-tabs > li {

float: left;

margin-bottom: -1px;

}

.nav-tabs > li > a {

margin-right: 2px;

line-height: 1.42857143;

border: 1px solid transparent;

border-radius: 4px 4px 0 0;

}

.nav-tabs > li > a:hover {

border-color: #eee #eee #ddd;

}

.nav-tabs > li.active > a,

.nav-tabs > li.active > a:hover,

.nav-tabs > li.active > a:focus {

color: #555;

cursor: default;

background-color: #fff;

border: 1px solid #ddd;

border-bottom-color: transparent;

}

.nav-tabs.nav-justified {

width: 100%;

border-bottom: 0;

}

.nav-tabs.nav-justified > li {

float: none;

}

.nav-tabs.nav-justified > li > a {

margin-bottom: 5px;

text-align: center;

}

.nav-tabs.nav-justified > .dropdown .dropdown-menu {

top: auto;

left: auto;

}

@media (min-width: 768px) {

.nav-tabs.nav-justified > li {

display: table-cell;

width: 1%;

}

.nav-tabs.nav-justified > li > a {

margin-bottom: 0;

}

}

.nav-tabs.nav-justified > li > a {

margin-right: 0;

border-radius: 4px;

}

.nav-tabs.nav-justified > .active > a,

.nav-tabs.nav-justified > .active > a:hover,

.nav-tabs.nav-justified > .active > a:focus {

border: 1px solid #ddd;

}

@media (min-width: 768px) {

.nav-tabs.nav-justified > li > a {

border-bottom: 1px solid #ddd;

border-radius: 4px 4px 0 0;

}

.nav-tabs.nav-justified > .active > a,

.nav-tabs.nav-justified > .active > a:hover,

.nav-tabs.nav-justified > .active > a:focus {

border-bottom-color: #fff;

}

}

.nav-pills > li {

float: left;

}

.nav-pills > li > a {

border-radius: 4px;

}

.nav-pills > li + li {

margin-left: 2px;

}

.nav-pills > li.active > a,

.nav-pills > li.active > a:hover,

.nav-pills > li.active > a:focus {

color: #fff;

background-color: #337ab7;

}

.nav-stacked > li {

float: none;

}

.nav-stacked > li + li {

margin-top: 2px;

margin-left: 0;

}

.nav-justified {

width: 100%;

}

.nav-justified > li {

float: none;

}

.nav-justified > li > a {

margin-bottom: 5px;

text-align: center;

}

.nav-justified > .dropdown .dropdown-menu {

top: auto;

left: auto;

}

@media (min-width: 768px) {

.nav-justified > li {

display: table-cell;

width: 1%;

}

.nav-justified > li > a {

margin-bottom: 0;

}

}

.nav-tabs-justified {

border-bottom: 0;

}

.nav-tabs-justified > li > a {

margin-right: 0;

border-radius: 4px;

}

.nav-tabs-justified > .active > a,

.nav-tabs-justified > .active > a:hover,

.nav-tabs-justified > .active > a:focus {

border: 1px solid #ddd;

}

@media (min-width: 768px) {

.nav-tabs-justified > li > a {

border-bottom: 1px solid #ddd;

border-radius: 4px 4px 0 0;

}

.nav-tabs-justified > .active > a,

.nav-tabs-justified > .active > a:hover,

.nav-tabs-justified > .active > a:focus {

border-bottom-color: #fff;

}

}

.tab-content > .tab-pane {

display: none;

}

.tab-content > .active {

display: block;

}

.nav-tabs .dropdown-menu {

margin-top: -1px;

border-top-left-radius: 0;

border-top-right-radius: 0;

}

.navbar {

position: relative;

min-height: 50px;

margin-bottom: 20px;

border: 1px solid transparent;

}

@media (min-width: 768px) {

.navbar {

border-radius: 4px;

}

}

@media (min-width: 768px) {

.navbar-header {

float: left;

}

}

.navbar-collapse {

padding-right: 15px;

padding-left: 15px;

overflow-x: visible;

-webkit-overflow-scrolling: touch;

border-top: 1px solid transparent;

-webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, .1);

box-shadow: inset 0 1px 0 rgba(255, 255, 255, .1);

}

.navbar-collapse.in {

overflow-y: auto;

}

@media (min-width: 768px) {

.navbar-collapse {

width: auto;

border-top: 0;

-webkit-box-shadow: none;

box-shadow: none;

}

.navbar-collapse.collapse {

display: block !important;

height: auto !important;

padding-bottom: 0;

overflow: visible !important;

}

.navbar-collapse.in {

overflow-y: visible;

}

.navbar-fixed-top .navbar-collapse,

.navbar-static-top .navbar-collapse,

.navbar-fixed-bottom .navbar-collapse {

padding-right: 0;

padding-left: 0;

}

}

.navbar-fixed-top .navbar-collapse,

.navbar-fixed-bottom .navbar-collapse {

max-height: 340px;

}

@media (max-device-width: 480px) and (orientation: landscape) {

.navbar-fixed-top .navbar-collapse,

.navbar-fixed-bottom .navbar-collapse {

max-height: 200px;

}

}

.container > .navbar-header,

.container-fluid > .navbar-header,

.container > .navbar-collapse,

.container-fluid > .navbar-collapse {

margin-right: -15px;

margin-left: -15px;

}

@media (min-width: 768px) {

.container > .navbar-header,

.container-fluid > .navbar-header,

.container > .navbar-collapse,

.container-fluid > .navbar-collapse {

margin-right: 0;

margin-left: 0;

}

}

.navbar-static-top {

z-index: 1000;

border-width: 0 0 1px;

}

@media (min-width: 768px) {

.navbar-static-top {

border-radius: 0;

}

}

.navbar-fixed-top,

.navbar-fixed-bottom {

position: fixed;

right: 0;

left: 0;

z-index: 1030;

}

@media (min-width: 768px) {

.navbar-fixed-top,

.navbar-fixed-bottom {

border-radius: 0;

}

}

.navbar-fixed-top {

top: 0;

border-width: 0 0 1px;

}

.navbar-fixed-bottom {

bottom: 0;

margin-bottom: 0;

border-width: 1px 0 0;

}

.navbar-brand {

float: left;

height: 50px;

padding: 15px 15px;

font-size: 18px;

line-height: 20px;

}

.navbar-brand:hover,

.navbar-brand:focus {

text-decoration: none;

}

.navbar-brand > img {

display: block;

}

@media (min-width: 768px) {

.navbar > .container .navbar-brand,

.navbar > .container-fluid .navbar-brand {

margin-left: -15px;

}

}

.navbar-toggle {

position: relative;

float: right;

padding: 9px 10px;

margin-top: 8px;

margin-right: 15px;

margin-bottom: 8px;

background-color: transparent;

background-image: none;

border: 1px solid transparent;

border-radius: 4px;

}

.navbar-toggle:focus {

outline: 0;

}

.navbar-toggle .icon-bar {

display: block;

width: 22px;

height: 2px;

border-radius: 1px;

}

.navbar-toggle .icon-bar + .icon-bar {

margin-top: 4px;

}

@media (min-width: 768px) {

.navbar-toggle {

display: none;

}

}

.navbar-nav {

margin: 7.5px -15px;

}

.navbar-nav > li > a {

padding-top: 10px;

padding-bottom: 10px;

line-height: 20px;

}

@media (max-width: 767px) {

.navbar-nav .open .dropdown-menu {

position: static;

float: none;

width: auto;

margin-top: 0;

background-color: transparent;

border: 0;

-webkit-box-shadow: none;

box-shadow: none;

}

.navbar-nav .open .dropdown-menu > li > a,

.navbar-nav .open .dropdown-menu .dropdown-header {

padding: 5px 15px 5px 25px;

}

.navbar-nav .open .dropdown-menu > li > a {

line-height: 20px;

}

.navbar-nav .open .dropdown-menu > li > a:hover,

.navbar-nav .open .dropdown-menu > li > a:focus {

background-image: none;

}

}

@media (min-width: 768px) {

.navbar-nav {

float: left;

margin: 0;

}

.navbar-nav > li {

float: left;

}

.navbar-nav > li > a {

padding-top: 15px;

padding-bottom: 15px;

}

}

.navbar-form {

padding: 10px 15px;

margin-top: 8px;

margin-right: -15px;

margin-bottom: 8px;

margin-left: -15px;

border-top: 1px solid transparent;

border-bottom: 1px solid transparent;

-webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, .1), 0 1px 0 rgba(255, 255, 255, .1);

box-shadow: inset 0 1px 0 rgba(255, 255, 255, .1), 0 1px 0 rgba(255, 255, 255, .1);

}

@media (min-width: 768px) {

.navbar-form .form-group {

display: inline-block;

margin-bottom: 0;

vertical-align: middle;

}

.navbar-form .form-control {

display: inline-block;

width: auto;

vertical-align: middle;

}

.navbar-form .form-control-static {

display: inline-block;

}

.navbar-form .input-group {

display: inline-table;

vertical-align: middle;

}

.navbar-form .input-group .input-group-addon,

.navbar-form .input-group .input-group-btn,

.navbar-form .input-group .form-control {

width: auto;

}

.navbar-form .input-group > .form-control {

width: 100%;

}

.navbar-form .control-label {

margin-bottom: 0;

vertical-align: middle;

}

.navbar-form .radio,

.navbar-form .checkbox {

display: inline-block;

margin-top: 0;

margin-bottom: 0;

vertical-align: middle;

}

.navbar-form .radio label,

.navbar-form .checkbox label {

padding-left: 0;

}

.navbar-form .radio input[type="radio"],

.navbar-form .checkbox input[type="checkbox"] {

position: relative;

margin-left: 0;

}

.navbar-form .has-feedback .form-control-feedback {

top: 0;

}

}

@media (max-width: 767px) {

.navbar-form .form-group {

margin-bottom: 5px;

}

.navbar-form .form-group:last-child {

margin-bottom: 0;

}

}

@media (min-width: 768px) {

.navbar-form {

width: auto;

padding-top: 0;

padding-bottom: 0;

margin-right: 0;

margin-left: 0;

border: 0;

-webkit-box-shadow: none;

box-shadow: none;

}

}

.navbar-nav > li > .dropdown-menu {

margin-top: 0;

border-top-left-radius: 0;

border-top-right-radius: 0;

}

.navbar-fixed-bottom .navbar-nav > li > .dropdown-menu {

margin-bottom: 0;

border-top-left-radius: 4px;

border-top-right-radius: 4px;

border-bottom-right-radius: 0;

border-bottom-left-radius: 0;

}

.navbar-btn {

margin-top: 8px;

margin-bottom: 8px;

}

.navbar-btn.btn-sm {

margin-top: 10px;

margin-bottom: 10px;

}

.navbar-btn.btn-xs {

margin-top: 14px;

margin-bottom: 14px;

}

.navbar-text {

margin-top: 15px;

margin-bottom: 15px;

}

@media (min-width: 768px) {

.navbar-text {

float: left;

margin-right: 15px;

margin-left: 15px;

}

}

@media (min-width: 768px) {

.navbar-left {

float: left !important;

}

.navbar-right {

float: right !important;

margin-right: -15px;

}

.navbar-right ~ .navbar-right {

margin-right: 0;

}

}

.navbar-default {

background-color: #f8f8f8;

border-color: #e7e7e7;

}

.navbar-default .navbar-brand {

color: #777;

}

.navbar-default .navbar-brand:hover,

.navbar-default .navbar-brand:focus {

color: #5e5e5e;

background-color: transparent;

}

.navbar-default .navbar-text {

color: #777;

}

.navbar-default .navbar-nav > li > a {

color: #777;

}

.navbar-default .navbar-nav > li > a:hover,

.navbar-default .navbar-nav > li > a:focus {

color: #333;

background-color: transparent;

}

.navbar-default .navbar-nav > .active > a,

.navbar-default .navbar-nav > .active > a:hover,

.navbar-default .navbar-nav > .active > a:focus {

color: #555;

background-color: #e7e7e7;

}

.navbar-default .navbar-nav > .disabled > a,

.navbar-default .navbar-nav > .disabled > a:hover,

.navbar-default .navbar-nav > .disabled > a:focus {

color: #ccc;

background-color: transparent;

}

.navbar-default .navbar-toggle {

border-color: #ddd;

}

.navbar-default .navbar-toggle:hover,

.navbar-default .navbar-toggle:focus {

background-color: #ddd;

}

.navbar-default .navbar-toggle .icon-bar {

background-color: #888;

}

.navbar-default .navbar-collapse,

.navbar-default .navbar-form {

border-color: #e7e7e7;

}

.navbar-default .navbar-nav > .open > a,

.navbar-default .navbar-nav > .open > a:hover,

.navbar-default .navbar-nav > .open > a:focus {

color: #555;

background-color: #e7e7e7;

}

@media (max-width: 767px) {

.navbar-default .navbar-nav .open .dropdown-menu > li > a {

color: #777;

}

.navbar-default .navbar-nav .open .dropdown-menu > li > a:hover,

.navbar-default .navbar-nav .open .dropdown-menu > li > a:focus {

color: #333;

background-color: transparent;

}

.navbar-default .navbar-nav .open .dropdown-menu > .active > a,

.navbar-default .navbar-nav .open .dropdown-menu > .active > a:hover,

.navbar-default .navbar-nav .open .dropdown-menu > .active > a:focus {

color: #555;

background-color: #e7e7e7;

}

.navbar-default .navbar-nav .open .dropdown-menu > .disabled > a,

.navbar-default .navbar-nav .open .dropdown-menu > .disabled > a:hover,

.navbar-default .navbar-nav .open .dropdown-menu > .disabled > a:focus {

color: #ccc;

background-color: transparent;

}

}

.navbar-default .navbar-link {

color: #777;

}

.navbar-default .navbar-link:hover {

color: #333;

}

.navbar-default .btn-link {

color: #777;

}

.navbar-default .btn-link:hover,

.navbar-default .btn-link:focus {

color: #333;

}

.navbar-default .btn-link[disabled]:hover,

fieldset[disabled] .navbar-default .btn-link:hover,

.navbar-default .btn-link[disabled]:focus,

fieldset[disabled] .navbar-default .btn-link:focus {

color: #ccc;

}

.navbar-inverse {

background-color: #222;

border-color: #080808;

}

.navbar-inverse .navbar-brand {

color: #9d9d9d;

}

.navbar-inverse .navbar-brand:hover,

.navbar-inverse .navbar-brand:focus {

color: #fff;

background-color: transparent;

}

.navbar-inverse .navbar-text {

color: #9d9d9d;

}

.navbar-inverse .navbar-nav > li > a {

color: #9d9d9d;

}

.navbar-inverse .navbar-nav > li > a:hover,

.navbar-inverse .navbar-nav > li > a:focus {

color: #fff;

background-color: transparent;

}

.navbar-inverse .navbar-nav > .active > a,

.navbar-inverse .navbar-nav > .active > a:hover,

.navbar-inverse .navbar-nav > .active > a:focus {

color: #fff;

background-color: #080808;

}

.navbar-inverse .navbar-nav > .disabled > a,

.navbar-inverse .navbar-nav > .disabled > a:hover,

.navbar-inverse .navbar-nav > .disabled > a:focus {

color: #444;

background-color: transparent;

}

.navbar-inverse .navbar-toggle {

border-color: #333;

}

.navbar-inverse .navbar-toggle:hover,

.navbar-inverse .navbar-toggle:focus {

background-color: #333;

}

.navbar-inverse .navbar-toggle .icon-bar {

background-color: #fff;

}

.navbar-inverse .navbar-collapse,

.navbar-inverse .navbar-form {

border-color: #101010;

}

.navbar-inverse .navbar-nav > .open > a,

.navbar-inverse .navbar-nav > .open > a:hover,

.navbar-inverse .navbar-nav > .open > a:focus {

color: #fff;

background-color: #080808;

}

@media (max-width: 767px) {

.navbar-inverse .navbar-nav .open .dropdown-menu > .dropdown-header {

border-color: #080808;

}

.navbar-inverse .navbar-nav .open .dropdown-menu .divider {

background-color: #080808;

}

.navbar-inverse .navbar-nav .open .dropdown-menu > li > a {

color: #9d9d9d;

}

.navbar-inverse .navbar-nav .open .dropdown-menu > li > a:hover,

.navbar-inverse .navbar-nav .open .dropdown-menu > li > a:focus {

color: #fff;

background-color: transparent;

}

.navbar-inverse .navbar-nav .open .dropdown-menu > .active > a,

.navbar-inverse .navbar-nav .open .dropdown-menu > .active > a:hover,

.navbar-inverse .navbar-nav .open .dropdown-menu > .active > a:focus {

color: #fff;

background-color: #080808;

}

.navbar-inverse .navbar-nav .open .dropdown-menu > .disabled > a,

.navbar-inverse .navbar-nav .open .dropdown-menu > .disabled > a:hover,

.navbar-inverse .navbar-nav .open .dropdown-menu > .disabled > a:focus {

color: #444;

background-color: transparent;

}

}

.navbar-inverse .navbar-link {

color: #9d9d9d;

}

.navbar-inverse .navbar-link:hover {

color: #fff;

}

.navbar-inverse .btn-link {

color: #9d9d9d;

}

.navbar-inverse .btn-link:hover,

.navbar-inverse .btn-link:focus {

color: #fff;

}

.navbar-inverse .btn-link[disabled]:hover,

fieldset[disabled] .navbar-inverse .btn-link:hover,

.navbar-inverse .btn-link[disabled]:focus,

fieldset[disabled] .navbar-inverse .btn-link:focus {

color: #444;

}

.breadcrumb {

padding: 8px 15px;

margin-bottom: 20px;

list-style: none;

background-color: #f5f5f5;

border-radius: 4px;

}

.breadcrumb > li {

display: inline-block;

}

.breadcrumb > li + li:before {

padding: 0 5px;

color: #ccc;

content: "/\00a0";

}

.breadcrumb > .active {

color: #777;

}

.pagination {

display: inline-block;

padding-left: 0;

margin: 20px 0;

border-radius: 4px;

}

.pagination > li {

display: inline;

}

.pagination > li > a,

.pagination > li > span {

position: relative;

float: left;

padding: 6px 12px;

margin-left: -1px;

line-height: 1.42857143;

color: #337ab7;

text-decoration: none;

background-color: #fff;

border: 1px solid #ddd;

}

.pagination > li:first-child > a,

.pagination > li:first-child > span {

margin-left: 0;

border-top-left-radius: 4px;

border-bottom-left-radius: 4px;

}

.pagination > li:last-child > a,

.pagination > li:last-child > span {

border-top-right-radius: 4px;

border-bottom-right-radius: 4px;

}

.pagination > li > a:hover,

.pagination > li > span:hover,

.pagination > li > a:focus,

.pagination > li > span:focus {

z-index: 2;

color: #23527c;

background-color: #eee;

border-color: #ddd;

}

.pagination > .active > a,

.pagination > .active > span,

.pagination > .active > a:hover,

.pagination > .active > span:hover,

.pagination > .active > a:focus,

.pagination > .active > span:focus {

z-index: 3;

color: #fff;

cursor: default;

background-color: #337ab7;

border-color: #337ab7;

}

.pagination > .disabled > span,

.pagination > .disabled > span:hover,

.pagination > .disabled > span:focus,

.pagination > .disabled > a,

.pagination > .disabled > a:hover,

.pagination > .disabled > a:focus {

color: #777;

cursor: not-allowed;

background-color: #fff;

border-color: #ddd;

}

.pagination-lg > li > a,

.pagination-lg > li > span {

padding: 10px 16px;

font-size: 18px;

line-height: 1.3333333;

}

.pagination-lg > li:first-child > a,

.pagination-lg > li:first-child > span {

border-top-left-radius: 6px;

border-bottom-left-radius: 6px;

}

.pagination-lg > li:last-child > a,

.pagination-lg > li:last-child > span {

border-top-right-radius: 6px;

border-bottom-right-radius: 6px;

}

.pagination-sm > li > a,

.pagination-sm > li > span {

padding: 5px 10px;

font-size: 12px;

line-height: 1.5;

}

.pagination-sm > li:first-child > a,

.pagination-sm > li:first-child > span {

border-top-left-radius: 3px;

border-bottom-left-radius: 3px;

}

.pagination-sm > li:last-child > a,

.pagination-sm > li:last-child > span {

border-top-right-radius: 3px;

border-bottom-right-radius: 3px;

}

.pager {

padding-left: 0;

margin: 20px 0;

text-align: center;

list-style: none;

}

.pager li {

display: inline;

}

.pager li > a,

.pager li > span {

display: inline-block;

padding: 5px 14px;

background-color: #fff;

border: 1px solid #ddd;

border-radius: 15px;

}

.pager li > a:hover,

.pager li > a:focus {

text-decoration: none;

background-color: #eee;

}

.pager .next > a,

.pager .next > span {

float: right;

}

.pager .previous > a,

.pager .previous > span {

float: left;

}

.pager .disabled > a,

.pager .disabled > a:hover,

.pager .disabled > a:focus,

.pager .disabled > span {

color: #777;

cursor: not-allowed;

background-color: #fff;

}

.label {

display: inline;

padding: .2em .6em .3em;

font-size: 75%;

font-weight: bold;

line-height: 1;

color: #fff;

text-align: center;

white-space: nowrap;

vertical-align: baseline;

border-radius: .25em;

}

a.label:hover,

a.label:focus {

color: #fff;

text-decoration: none;

cursor: pointer;

}

.label:empty {

display: none;

}

.btn .label {

position: relative;

top: -1px;

}

.label-default {

background-color: #777;

}

.label-default[href]:hover,

.label-default[href]:focus {

background-color: #5e5e5e;

}

.label-primary {

background-color: #337ab7;

}

.label-primary[href]:hover,

.label-primary[href]:focus {

background-color: #286090;

}

.label-success {

background-color: #5cb85c;

}

.label-success[href]:hover,

.label-success[href]:focus {

background-color: #449d44;

}

.label-info {

background-color: #5bc0de;

}

.label-info[href]:hover,

.label-info[href]:focus {

background-color: #31b0d5;

}

.label-warning {

background-color: #f0ad4e;

}

.label-warning[href]:hover,

.label-warning[href]:focus {

background-color: #ec971f;

}

.label-danger {

background-color: #d9534f;

}

.label-danger[href]:hover,

.label-danger[href]:focus {

background-color: #c9302c;

}

.badge {

display: inline-block;

min-width: 10px;

padding: 3px 7px;

font-size: 12px;

font-weight: bold;

line-height: 1;

color: #fff;

text-align: center;

white-space: nowrap;

vertical-align: middle;

background-color: #777;

border-radius: 10px;

}

.badge:empty {

display: none;

}

.btn .badge {

position: relative;

top: -1px;

}

.btn-xs .badge,

.btn-group-xs > .btn .badge {

top: 0;

padding: 1px 5px;

}

a.badge:hover,

a.badge:focus {

color: #fff;

text-decoration: none;

cursor: pointer;

}

.list-group-item.active > .badge,

.nav-pills > .active > a > .badge {

color: #337ab7;

background-color: #fff;

}

.list-group-item > .badge {

float: right;

}

.list-group-item > .badge + .badge {

margin-right: 5px;

}

.nav-pills > li > a > .badge {

margin-left: 3px;

}

.jumbotron {

padding-top: 30px;

padding-bottom: 30px;

margin-bottom: 30px;

color: inherit;

background-color: #eee;

}

.jumbotron h1,

.jumbotron .h1 {

color: inherit;

}

.jumbotron p {

margin-bottom: 15px;

font-size: 21px;

font-weight: 200;

}

.jumbotron > hr {

border-top-color: #d5d5d5;

}

.container .jumbotron,

.container-fluid .jumbotron {

padding-right: 15px;

padding-left: 15px;

border-radius: 6px;

}

.jumbotron .container {

max-width: 100%;

}

@media screen and (min-width: 768px) {

.jumbotron {

padding-top: 48px;

padding-bottom: 48px;

}

.container .jumbotron,

.container-fluid .jumbotron {

padding-right: 60px;

padding-left: 60px;

}

.jumbotron h1,

.jumbotron .h1 {

font-size: 63px;

}

}

.thumbnail {

display: block;

padding: 4px;

margin-bottom: 20px;

line-height: 1.42857143;

background-color: #fff;

border: 1px solid #ddd;

border-radius: 4px;

-webkit-transition: border .2s ease-in-out;

-o-transition: border .2s ease-in-out;

transition: border .2s ease-in-out;

}

.thumbnail > img,

.thumbnail a > img {

margin-right: auto;

margin-left: auto;

}

a.thumbnail:hover,

a.thumbnail:focus,

a.thumbnail.active {

border-color: #337ab7;

}

.thumbnail .caption {

padding: 9px;

color: #333;

}

.alert {

padding: 15px;

margin-bottom: 20px;

border: 1px solid transparent;

border-radius: 4px;

}

.alert h4 {

margin-top: 0;

color: inherit;

}

.alert .alert-link {

font-weight: bold;

}

.alert > p,

.alert > ul {

margin-bottom: 0;

}

.alert > p + p {

margin-top: 5px;

}

.alert-dismissable,

.alert-dismissible {

padding-right: 35px;

}

.alert-dismissable .close,

.alert-dismissible .close {

position: relative;

top: -2px;

right: -21px;

color: inherit;

}

.alert-success {

color: #3c763d;

background-color: #dff0d8;

border-color: #d6e9c6;

}

.alert-success hr {

border-top-color: #c9e2b3;

}

.alert-success .alert-link {

color: #2b542c;

}

.alert-info {

color: #31708f;

background-color: #d9edf7;

border-color: #bce8f1;

}

.alert-info hr {

border-top-color: #a6e1ec;

}

.alert-info .alert-link {

color: #245269;

}

.alert-warning {

color: #8a6d3b;

background-color: #fcf8e3;

border-color: #faebcc;

}

.alert-warning hr {

border-top-color: #f7e1b5;

}

.alert-warning .alert-link {

color: #66512c;

}

.alert-danger {

color: #a94442;

background-color: #f2dede;

border-color: #ebccd1;

}

.alert-danger hr {

border-top-color: #e4b9c0;

}

.alert-danger .alert-link {

color: #843534;

}

@-webkit-keyframes progress-bar-stripes {

from {

background-position: 40px 0;

}

to {

background-position: 0 0;

}

}

@-o-keyframes progress-bar-stripes {

from {

background-position: 40px 0;

}

to {

background-position: 0 0;

}

}

@keyframes progress-bar-stripes {

from {

background-position: 40px 0;

}

to {

background-position: 0 0;

}

}

.progress {

height: 20px;

margin-bottom: 20px;

overflow: hidden;

background-color: #f5f5f5;

border-radius: 4px;

-webkit-box-shadow: inset 0 1px 2px rgba(0, 0, 0, .1);

box-shadow: inset 0 1px 2px rgba(0, 0, 0, .1);

}

.progress-bar {

float: left;

width: 0;

height: 100%;

font-size: 12px;

line-height: 20px;

color: #fff;

text-align: center;

background-color: #337ab7;

-webkit-box-shadow: inset 0 -1px 0 rgba(0, 0, 0, .15);

box-shadow: inset 0 -1px 0 rgba(0, 0, 0, .15);

-webkit-transition: width .6s ease;

-o-transition: width .6s ease;

transition: width .6s ease;

}

.progress-striped .progress-bar,

.progress-bar-striped {

background-image: -webkit-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: -o-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

-webkit-background-size: 40px 40px;

background-size: 40px 40px;

}

.progress.active .progress-bar,

.progress-bar.active {

-webkit-animation: progress-bar-stripes 2s linear infinite;

-o-animation: progress-bar-stripes 2s linear infinite;

animation: progress-bar-stripes 2s linear infinite;

}

.progress-bar-success {

background-color: #5cb85c;

}

.progress-striped .progress-bar-success {

background-image: -webkit-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: -o-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

}

.progress-bar-info {

background-color: #5bc0de;

}

.progress-striped .progress-bar-info {

background-image: -webkit-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: -o-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

}

.progress-bar-warning {

background-color: #f0ad4e;

}

.progress-striped .progress-bar-warning {

background-image: -webkit-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: -o-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

}

.progress-bar-danger {

background-color: #d9534f;

}

.progress-striped .progress-bar-danger {

background-image: -webkit-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: -o-linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

background-image: linear-gradient(45deg, rgba(255, 255, 255, .15) 25%, transparent 25%, transparent 50%, rgba(255, 255, 255, .15) 50%, rgba(255, 255, 255, .15) 75%, transparent 75%, transparent);

}

.media {

margin-top: 15px;

}

.media:first-child {

margin-top: 0;

}

.media,

.media-body {

overflow: hidden;

zoom: 1;

}

.media-body {

width: 10000px;

}

.media-object {

display: block;

}

.media-object.img-thumbnail {

max-width: none;

}

.media-right,

.media > .pull-right {

padding-left: 10px;

}

.media-left,

.media > .pull-left {

padding-right: 10px;

}

.media-left,

.media-right,

.media-body {

display: table-cell;

vertical-align: top;

}

.media-middle {

vertical-align: middle;

}

.media-bottom {

vertical-align: bottom;

}

.media-heading {

margin-top: 0;

margin-bottom: 5px;

}

.media-list {

padding-left: 0;

list-style: none;

}

.list-group {

padding-left: 0;

margin-bottom: 20px;

}

.list-group-item {

position: relative;

display: block;

padding: 10px 15px;

margin-bottom: -1px;

background-color: #fff;

border: 1px solid #ddd;

}

.list-group-item:first-child {

border-top-left-radius: 4px;

border-top-right-radius: 4px;

}

.list-group-item:last-child {

margin-bottom: 0;

border-bottom-right-radius: 4px;

border-bottom-left-radius: 4px;

}

a.list-group-item,

button.list-group-item {

color: #555;

}

a.list-group-item .list-group-item-heading,

button.list-group-item .list-group-item-heading {

color: #333;

}

a.list-group-item:hover,

button.list-group-item:hover,

a.list-group-item:focus,

button.list-group-item:focus {

color: #555;

text-decoration: none;

background-color: #f5f5f5;

}

button.list-group-item {

width: 100%;

text-align: left;

}

.list-group-item.disabled,

.list-group-item.disabled:hover,

.list-group-item.disabled:focus {

color: #777;

cursor: not-allowed;

background-color: #eee;

}

.list-group-item.disabled .list-group-item-heading,

.list-group-item.disabled:hover .list-group-item-heading,

.list-group-item.disabled:focus .list-group-item-heading {

color: inherit;

}

.list-group-item.disabled .list-group-item-text,

.list-group-item.disabled:hover .list-group-item-text,

.list-group-item.disabled:focus .list-group-item-text {

color: #777;

}

.list-group-item.active,

.list-group-item.active:hover,

.list-group-item.active:focus {

z-index: 2;

color: #fff;

background-color: #337ab7;

border-color: #337ab7;

}

.list-group-item.active .list-group-item-heading,

.list-group-item.active:hover .list-group-item-heading,

.list-group-item.active:focus .list-group-item-heading,

.list-group-item.active .list-group-item-heading > small,

.list-group-item.active:hover .list-group-item-heading > small,

.list-group-item.active:focus .list-group-item-heading > small,

.list-group-item.active .list-group-item-heading > .small,

.list-group-item.active:hover .list-group-item-heading > .small,

.list-group-item.active:focus .list-group-item-heading > .small {

color: inherit;

}

.list-group-item.active .list-group-item-text,

.list-group-item.active:hover .list-group-item-text,

.list-group-item.active:focus .list-group-item-text {

color: #c7ddef;

}

.list-group-item-success {

color: #3c763d;

background-color: #dff0d8;

}

a.list-group-item-success,

button.list-group-item-success {

color: #3c763d;

}

a.list-group-item-success .list-group-item-heading,

button.list-group-item-success .list-group-item-heading {

color: inherit;

}

a.list-group-item-success:hover,

button.list-group-item-success:hover,

a.list-group-item-success:focus,

button.list-group-item-success:focus {

color: #3c763d;

background-color: #d0e9c6;

}

a.list-group-item-success.active,

button.list-group-item-success.active,

a.list-group-item-success.active:hover,

button.list-group-item-success.active:hover,

a.list-group-item-success.active:focus,

button.list-group-item-success.active:focus {

color: #fff;

background-color: #3c763d;

border-color: #3c763d;

}

.list-group-item-info {

color: #31708f;

background-color: #d9edf7;

}

a.list-group-item-info,

button.list-group-item-info {

color: #31708f;

}

a.list-group-item-info .list-group-item-heading,

button.list-group-item-info .list-group-item-heading {

color: inherit;

}

a.list-group-item-info:hover,

button.list-group-item-info:hover,

a.list-group-item-info:focus,

button.list-group-item-info:focus {

color: #31708f;

background-color: #c4e3f3;

}

a.list-group-item-info.active,

button.list-group-item-info.active,

a.list-group-item-info.active:hover,

button.list-group-item-info.active:hover,

a.list-group-item-info.active:focus,

button.list-group-item-info.active:focus {

color: #fff;

background-color: #31708f;

border-color: #31708f;

}

.list-group-item-warning {

color: #8a6d3b;

background-color: #fcf8e3;

}

a.list-group-item-warning,

button.list-group-item-warning {

color: #8a6d3b;

}

a.list-group-item-warning .list-group-item-heading,

button.list-group-item-warning .list-group-item-heading {

color: inherit;

}

a.list-group-item-warning:hover,

button.list-group-item-warning:hover,

a.list-group-item-warning:focus,

button.list-group-item-warning:focus {

color: #8a6d3b;

background-color: #faf2cc;

}

a.list-group-item-warning.active,

button.list-group-item-warning.active,

a.list-group-item-warning.active:hover,

button.list-group-item-warning.active:hover,

a.list-group-item-warning.active:focus,

button.list-group-item-warning.active:focus {

color: #fff;

background-color: #8a6d3b;

border-color: #8a6d3b;

}

.list-group-item-danger {

color: #a94442;

background-color: #f2dede;

}

a.list-group-item-danger,

button.list-group-item-danger {

color: #a94442;

}

a.list-group-item-danger .list-group-item-heading,

button.list-group-item-danger .list-group-item-heading {

color: inherit;

}

a.list-group-item-danger:hover,

button.list-group-item-danger:hover,

a.list-group-item-danger:focus,

button.list-group-item-danger:focus {

color: #a94442;

background-color: #ebcccc;

}

a.list-group-item-danger.active,

button.list-group-item-danger.active,

a.list-group-item-danger.active:hover,

button.list-group-item-danger.active:hover,

a.list-group-item-danger.active:focus,

button.list-group-item-danger.active:focus {

color: #fff;

background-color: #a94442;

border-color: #a94442;

}

.list-group-item-heading {

margin-top: 0;

margin-bottom: 5px;

}

.list-group-item-text {

margin-bottom: 0;

line-height: 1.3;

}

.panel {

margin-bottom: 20px;

background-color: #fff;

border: 1px solid transparent;

border-radius: 4px;

-webkit-box-shadow: 0 1px 1px rgba(0, 0, 0, .05);

box-shadow: 0 1px 1px rgba(0, 0, 0, .05);

}

.panel-body {

padding: 15px;

}

.panel-heading {

padding: 10px 15px;

border-bottom: 1px solid transparent;

border-top-left-radius: 3px;

border-top-right-radius: 3px;

}

.panel-heading > .dropdown .dropdown-toggle {

color: inherit;

}

.panel-title {

margin-top: 0;

margin-bottom: 0;

font-size: 16px;

color: inherit;

}

.panel-title > a,

.panel-title > small,

.panel-title > .small,

.panel-title > small > a,

.panel-title > .small > a {

color: inherit;

}

.panel-footer {

padding: 10px 15px;

background-color: #f5f5f5;

border-top: 1px solid #ddd;

border-bottom-right-radius: 3px;

border-bottom-left-radius: 3px;

}

.panel > .list-group,

.panel > .panel-collapse > .list-group {

margin-bottom: 0;

}

.panel > .list-group .list-group-item,

.panel > .panel-collapse > .list-group .list-group-item {

border-width: 1px 0;

border-radius: 0;

}

.panel > .list-group:first-child .list-group-item:first-child,

.panel > .panel-collapse > .list-group:first-child .list-group-item:first-child {

border-top: 0;

border-top-left-radius: 3px;

border-top-right-radius: 3px;

}

.panel > .list-group:last-child .list-group-item:last-child,

.panel > .panel-collapse > .list-group:last-child .list-group-item:last-child {

border-bottom: 0;

border-bottom-right-radius: 3px;

border-bottom-left-radius: 3px;

}

.panel > .panel-heading + .panel-collapse > .list-group .list-group-item:first-child {

border-top-left-radius: 0;

border-top-right-radius: 0;

}

.panel-heading + .list-group .list-group-item:first-child {

border-top-width: 0;

}

.list-group + .panel-footer {

border-top-width: 0;

}

.panel > .table,

.panel > .table-responsive > .table,

.panel > .panel-collapse > .table {

margin-bottom: 0;

}

.panel > .table caption,

.panel > .table-responsive > .table caption,

.panel > .panel-collapse > .table caption {

padding-right: 15px;

padding-left: 15px;

}

.panel > .table:first-child,

.panel > .table-responsive:first-child > .table:first-child {

border-top-left-radius: 3px;

border-top-right-radius: 3px;

}

.panel > .table:first-child > thead:first-child > tr:first-child,

.panel > .table-responsive:first-child > .table:first-child > thead:first-child > tr:first-child,

.panel > .table:first-child > tbody:first-child > tr:first-child,

.panel > .table-responsive:first-child > .table:first-child > tbody:first-child > tr:first-child {

border-top-left-radius: 3px;

border-top-right-radius: 3px;

}

.panel > .table:first-child > thead:first-child > tr:first-child td:first-child,

.panel > .table-responsive:first-child > .table:first-child > thead:first-child > tr:first-child td:first-child,

.panel > .table:first-child > tbody:first-child > tr:first-child td:first-child,

.panel > .table-responsive:first-child > .table:first-child > tbody:first-child > tr:first-child td:first-child,

.panel > .table:first-child > thead:first-child > tr:first-child th:first-child,

.panel > .table-responsive:first-child > .table:first-child > thead:first-child > tr:first-child th:first-child,

.panel > .table:first-child > tbody:first-child > tr:first-child th:first-child,

.panel > .table-responsive:first-child > .table:first-child > tbody:first-child > tr:first-child th:first-child {

border-top-left-radius: 3px;

}

.panel > .table:first-child > thead:first-child > tr:first-child td:last-child,

.panel > .table-responsive:first-child > .table:first-child > thead:first-child > tr:first-child td:last-child,

.panel > .table:first-child > tbody:first-child > tr:first-child td:last-child,

.panel > .table-responsive:first-child > .table:first-child > tbody:first-child > tr:first-child td:last-child,

.panel > .table:first-child > thead:first-child > tr:first-child th:last-child,

.panel > .table-responsive:first-child > .table:first-child > thead:first-child > tr:first-child th:last-child,

.panel > .table:first-child > tbody:first-child > tr:first-child th:last-child,

.panel > .table-responsive:first-child > .table:first-child > tbody:first-child > tr:first-child th:last-child {

border-top-right-radius: 3px;

}

.panel > .table:last-child,

.panel > .table-responsive:last-child > .table:last-child {

border-bottom-right-radius: 3px;

border-bottom-left-radius: 3px;

}

.panel > .table:last-child > tbody:last-child > tr:last-child,

.panel > .table-responsive:last-child > .table:last-child > tbody:last-child > tr:last-child,

.panel > .table:last-child > tfoot:last-child > tr:last-child,

.panel > .table-responsive:last-child > .table:last-child > tfoot:last-child > tr:last-child {

border-bottom-right-radius: 3px;

border-bottom-left-radius: 3px;

}

.panel > .table:last-child > tbody:last-child > tr:last-child td:first-child,

.panel > .table-responsive:last-child > .table:last-child > tbody:last-child > tr:last-child td:first-child,

.panel > .table:last-child > tfoot:last-child > tr:last-child td:first-child,

.panel > .table-responsive:last-child > .table:last-child > tfoot:last-child > tr:last-child td:first-child,

.panel > .table:last-child > tbody:last-child > tr:last-child th:first-child,

.panel > .table-responsive:last-child > .table:last-child > tbody:last-child > tr:last-child th:first-child,

.panel > .table:last-child > tfoot:last-child > tr:last-child th:first-child,

.panel > .table-responsive:last-child > .table:last-child > tfoot:last-child > tr:last-child th:first-child {

border-bottom-left-radius: 3px;

}

.panel > .table:last-child > tbody:last-child > tr:last-child td:last-child,

.panel > .table-responsive:last-child > .table:last-child > tbody:last-child > tr:last-child td:last-child,

.panel > .table:last-child > tfoot:last-child > tr:last-child td:last-child,

.panel > .table-responsive:last-child > .table:last-child > tfoot:last-child > tr:last-child td:last-child,

.panel > .table:last-child > tbody:last-child > tr:last-child th:last-child,

.panel > .table-responsive:last-child > .table:last-child > tbody:last-child > tr:last-child th:last-child,

.panel > .table:last-child > tfoot:last-child > tr:last-child th:last-child,

.panel > .table-responsive:last-child > .table:last-child > tfoot:last-child > tr:last-child th:last-child {

border-bottom-right-radius: 3px;

}

.panel > .panel-body + .table,

.panel > .panel-body + .table-responsive,

.panel > .table + .panel-body,

.panel > .table-responsive + .panel-body {

border-top: 1px solid #ddd;

}

.panel > .table > tbody:first-child > tr:first-child th,

.panel > .table > tbody:first-child > tr:first-child td {

border-top: 0;

}

.panel > .table-bordered,

.panel > .table-responsive > .table-bordered {

border: 0;

}

.panel > .table-bordered > thead > tr > th:first-child,

.panel > .table-responsive > .table-bordered > thead > tr > th:first-child,

.panel > .table-bordered > tbody > tr > th:first-child,

.panel > .table-responsive > .table-bordered > tbody > tr > th:first-child,

.panel > .table-bordered > tfoot > tr > th:first-child,

.panel > .table-responsive > .table-bordered > tfoot > tr > th:first-child,

.panel > .table-bordered > thead > tr > td:first-child,

.panel > .table-responsive > .table-bordered > thead > tr > td:first-child,

.panel > .table-bordered > tbody > tr > td:first-child,

.panel > .table-responsive > .table-bordered > tbody > tr > td:first-child,

.panel > .table-bordered > tfoot > tr > td:first-child,

.panel > .table-responsive > .table-bordered > tfoot > tr > td:first-child {

border-left: 0;

}

.panel > .table-bordered > thead > tr > th:last-child,

.panel > .table-responsive > .table-bordered > thead > tr > th:last-child,

.panel > .table-bordered > tbody > tr > th:last-child,

.panel > .table-responsive > .table-bordered > tbody > tr > th:last-child,

.panel > .table-bordered > tfoot > tr > th:last-child,

.panel > .table-responsive > .table-bordered > tfoot > tr > th:last-child,

.panel > .table-bordered > thead > tr > td:last-child,

.panel > .table-responsive > .table-bordered > thead > tr > td:last-child,

.panel > .table-bordered > tbody > tr > td:last-child,

.panel > .table-responsive > .table-bordered > tbody > tr > td:last-child,

.panel > .table-bordered > tfoot > tr > td:last-child,

.panel > .table-responsive > .table-bordered > tfoot > tr > td:last-child {

border-right: 0;

}

.panel > .table-bordered > thead > tr:first-child > td,

.panel > .table-responsive > .table-bordered > thead > tr:first-child > td,

.panel > .table-bordered > tbody > tr:first-child > td,

.panel > .table-responsive > .table-bordered > tbody > tr:first-child > td,

.panel > .table-bordered > thead > tr:first-child > th,

.panel > .table-responsive > .table-bordered > thead > tr:first-child > th,

.panel > .table-bordered > tbody > tr:first-child > th,

.panel > .table-responsive > .table-bordered > tbody > tr:first-child > th {

border-bottom: 0;

}

.panel > .table-bordered > tbody > tr:last-child > td,

.panel > .table-responsive > .table-bordered > tbody > tr:last-child > td,

.panel > .table-bordered > tfoot > tr:last-child > td,

.panel > .table-responsive > .table-bordered > tfoot > tr:last-child > td,

.panel > .table-bordered > tbody > tr:last-child > th,

.panel > .table-responsive > .table-bordered > tbody > tr:last-child > th,

.panel > .table-bordered > tfoot > tr:last-child > th,

.panel > .table-responsive > .table-bordered > tfoot > tr:last-child > th {

border-bottom: 0;

}

.panel > .table-responsive {

margin-bottom: 0;

border: 0;

}

.panel-group {

margin-bottom: 20px;

}

.panel-group .panel {

margin-bottom: 0;

border-radius: 4px;

}

.panel-group .panel + .panel {

margin-top: 5px;

}

.panel-group .panel-heading {

border-bottom: 0;

}

.panel-group .panel-heading + .panel-collapse > .panel-body,

.panel-group .panel-heading + .panel-collapse > .list-group {

border-top: 1px solid #ddd;

}

.panel-group .panel-footer {

border-top: 0;

}

.panel-group .panel-footer + .panel-collapse .panel-body {

border-bottom: 1px solid #ddd;

}

.panel-default {

border-color: #ddd;

}

.panel-default > .panel-heading {

color: #333;

background-color: #f5f5f5;

border-color: #ddd;

}

.panel-default > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #ddd;

}

.panel-default > .panel-heading .badge {

color: #f5f5f5;

background-color: #333;

}

.panel-default > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #ddd;

}

.panel-primary {

border-color: #337ab7;

}

.panel-primary > .panel-heading {

color: #fff;

background-color: #337ab7;

border-color: #337ab7;

}

.panel-primary > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #337ab7;

}

.panel-primary > .panel-heading .badge {

color: #337ab7;

background-color: #fff;

}

.panel-primary > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #337ab7;

}

.panel-success {

border-color: #d6e9c6;

}

.panel-success > .panel-heading {

color: #3c763d;

background-color: #dff0d8;

border-color: #d6e9c6;

}

.panel-success > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #d6e9c6;

}

.panel-success > .panel-heading .badge {

color: #dff0d8;

background-color: #3c763d;

}

.panel-success > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #d6e9c6;

}

.panel-info {

border-color: #bce8f1;

}

.panel-info > .panel-heading {

color: #31708f;

background-color: #d9edf7;

border-color: #bce8f1;

}

.panel-info > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #bce8f1;

}

.panel-info > .panel-heading .badge {

color: #d9edf7;

background-color: #31708f;

}

.panel-info > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #bce8f1;

}

.panel-warning {

border-color: #faebcc;

}

.panel-warning > .panel-heading {

color: #8a6d3b;

background-color: #fcf8e3;

border-color: #faebcc;

}

.panel-warning > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #faebcc;

}

.panel-warning > .panel-heading .badge {

color: #fcf8e3;

background-color: #8a6d3b;

}

.panel-warning > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #faebcc;

}

.panel-danger {

border-color: #ebccd1;

}

.panel-danger > .panel-heading {

color: #a94442;

background-color: #f2dede;

border-color: #ebccd1;

}

.panel-danger > .panel-heading + .panel-collapse > .panel-body {

border-top-color: #ebccd1;

}

.panel-danger > .panel-heading .badge {

color: #f2dede;

background-color: #a94442;

}

.panel-danger > .panel-footer + .panel-collapse > .panel-body {

border-bottom-color: #ebccd1;

}

.embed-responsive {

position: relative;

display: block;

height: 0;

padding: 0;

overflow: hidden;

}

.embed-responsive .embed-responsive-item,

.embed-responsive iframe,

.embed-responsive embed,

.embed-responsive object,

.embed-responsive video {

position: absolute;

top: 0;

bottom: 0;

left: 0;

width: 100%;

height: 100%;

border: 0;

}

.embed-responsive-16by9 {

padding-bottom: 56.25%;

}

.embed-responsive-4by3 {

padding-bottom: 75%;

}

.well {

min-height: 20px;

padding: 19px;

margin-bottom: 20px;

background-color: #f5f5f5;

border: 1px solid #e3e3e3;

border-radius: 4px;

-webkit-box-shadow: inset 0 1px 1px rgba(0, 0, 0, .05);

box-shadow: inset 0 1px 1px rgba(0, 0, 0, .05);

}

.well blockquote {

border-color: #ddd;

border-color: rgba(0, 0, 0, .15);

}

.well-lg {

padding: 24px;

border-radius: 6px;

}

.well-sm {

padding: 9px;

border-radius: 3px;

}

.close {

float: right;

font-size: 21px;

font-weight: bold;

line-height: 1;

color: #000;

text-shadow: 0 1px 0 #fff;

filter: alpha(opacity=20);

opacity: .2;

}

.close:hover,

.close:focus {

color: #000;

text-decoration: none;

cursor: pointer;

filter: alpha(opacity=50);

opacity: .5;

}

button.close {

-webkit-appearance: none;

padding: 0;

cursor: pointer;

background: transparent;

border: 0;

}

.modal-open {

overflow: hidden;

}

.modal {

position: fixed;

top: 0;

right: 0;

bottom: 0;

left: 0;

z-index: 1050;

display: none;

overflow: hidden;

-webkit-overflow-scrolling: touch;

outline: 0;

}

.modal.fade .modal-dialog {

-webkit-transition: -webkit-transform .3s ease-out;

-o-transition: -o-transform .3s ease-out;

transition: transform .3s ease-out;

-webkit-transform: translate(0, -25%);

-ms-transform: translate(0, -25%);

-o-transform: translate(0, -25%);

transform: translate(0, -25%);

}

.modal.in .modal-dialog {

-webkit-transform: translate(0, 0);

-ms-transform: translate(0, 0);

-o-transform: translate(0, 0);

transform: translate(0, 0);

}

.modal-open .modal {

overflow-x: hidden;

overflow-y: auto;

}

.modal-dialog {

position: relative;

width: auto;

margin: 10px;

}

.modal-content {

position: relative;

background-color: #fff;

-webkit-background-clip: padding-box;

background-clip: padding-box;

border: 1px solid #999;

border: 1px solid rgba(0, 0, 0, .2);

border-radius: 6px;

outline: 0;

-webkit-box-shadow: 0 3px 9px rgba(0, 0, 0, .5);

box-shadow: 0 3px 9px rgba(0, 0, 0, .5);

}

.modal-backdrop {

position: fixed;

top: 0;

right: 0;

bottom: 0;

left: 0;

z-index: 1040;

background-color: #000;

}

.modal-backdrop.fade {

filter: alpha(opacity=0);

opacity: 0;

}

.modal-backdrop.in {

filter: alpha(opacity=50);

opacity: .5;

}

.modal-header {

padding: 15px;

border-bottom: 1px solid #e5e5e5;

}

.modal-header .close {

margin-top: -2px;

}

.modal-title {

margin: 0;

line-height: 1.42857143;

}

.modal-body {

position: relative;

padding: 15px;

}

.modal-footer {

padding: 15px;

text-align: right;

border-top: 1px solid #e5e5e5;

}

.modal-footer .btn + .btn {

margin-bottom: 0;

margin-left: 5px;

}

.modal-footer .btn-group .btn + .btn {

margin-left: -1px;

}

.modal-footer .btn-block + .btn-block {

margin-left: 0;

}

.modal-scrollbar-measure {

position: absolute;

top: -9999px;

width: 50px;

height: 50px;

overflow: scroll;

}

@media (min-width: 768px) {

.modal-dialog {

width: 600px;

margin: 30px auto;

}

.modal-content {

-webkit-box-shadow: 0 5px 15px rgba(0, 0, 0, .5);

box-shadow: 0 5px 15px rgba(0, 0, 0, .5);

}

.modal-sm {

width: 300px;

}

}

@media (min-width: 992px) {

.modal-lg {

width: 900px;

}

}

.tooltip {

position: absolute;

z-index: 1070;

display: block;

font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;

font-size: 12px;

font-style: normal;

font-weight: normal;

line-height: 1.42857143;

text-align: left;

text-align: start;

text-decoration: none;

text-shadow: none;

text-transform: none;

letter-spacing: normal;

word-break: normal;

word-spacing: normal;

word-wrap: normal;

white-space: normal;

filter: alpha(opacity=0);

opacity: 0;

line-break: auto;

}

.tooltip.in {

filter: alpha(opacity=90);

opacity: .9;

}

.tooltip.top {

padding: 5px 0;

margin-top: -3px;

}

.tooltip.right {

padding: 0 5px;

margin-left: 3px;

}

.tooltip.bottom {

padding: 5px 0;

margin-top: 3px;

}

.tooltip.left {

padding: 0 5px;

margin-left: -3px;

}

.tooltip-inner {

max-width: 200px;

padding: 3px 8px;

color: #fff;

text-align: center;

background-color: #000;

border-radius: 4px;

}

.tooltip-arrow {

position: absolute;

width: 0;

height: 0;

border-color: transparent;

border-style: solid;

}

.tooltip.top .tooltip-arrow {

bottom: 0;

left: 50%;

margin-left: -5px;

border-width: 5px 5px 0;

border-top-color: #000;

}

.tooltip.top-left .tooltip-arrow {

right: 5px;

bottom: 0;

margin-bottom: -5px;

border-width: 5px 5px 0;

border-top-color: #000;

}

.tooltip.top-right .tooltip-arrow {

bottom: 0;

left: 5px;

margin-bottom: -5px;

border-width: 5px 5px 0;

border-top-color: #000;

}

.tooltip.right .tooltip-arrow {

top: 50%;

left: 0;

margin-top: -5px;

border-width: 5px 5px 5px 0;

border-right-color: #000;

}

.tooltip.left .tooltip-arrow {

top: 50%;

right: 0;

margin-top: -5px;

border-width: 5px 0 5px 5px;

border-left-color: #000;

}

.tooltip.bottom .tooltip-arrow {

top: 0;

left: 50%;

margin-left: -5px;

border-width: 0 5px 5px;

border-bottom-color: #000;

}

.tooltip.bottom-left .tooltip-arrow {

top: 0;

right: 5px;

margin-top: -5px;

border-width: 0 5px 5px;

border-bottom-color: #000;

}

.tooltip.bottom-right .tooltip-arrow {

top: 0;

left: 5px;

margin-top: -5px;

border-width: 0 5px 5px;

border-bottom-color: #000;

}

.popover {

position: absolute;

top: 0;

left: 0;

z-index: 1060;

display: none;

max-width: 276px;

padding: 1px;

font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;

font-size: 14px;

font-style: normal;

font-weight: normal;

line-height: 1.42857143;

text-align: left;

text-align: start;

text-decoration: none;

text-shadow: none;

text-transform: none;

letter-spacing: normal;

word-break: normal;

word-spacing: normal;

word-wrap: normal;

white-space: normal;

background-color: #fff;

-webkit-background-clip: padding-box;

background-clip: padding-box;

border: 1px solid #ccc;

border: 1px solid rgba(0, 0, 0, .2);

border-radius: 6px;

-webkit-box-shadow: 0 5px 10px rgba(0, 0, 0, .2);

box-shadow: 0 5px 10px rgba(0, 0, 0, .2);

line-break: auto;

}

.popover.top {

margin-top: -10px;

}

.popover.right {

margin-left: 10px;

}

.popover.bottom {

margin-top: 10px;

}

.popover.left {

margin-left: -10px;

}

.popover-title {

padding: 8px 14px;

margin: 0;

font-size: 14px;

background-color: #f7f7f7;

border-bottom: 1px solid #ebebeb;

border-radius: 5px 5px 0 0;

}

.popover-content {

padding: 9px 14px;

}

.popover > .arrow,

.popover > .arrow:after {

position: absolute;

display: block;

width: 0;

height: 0;

border-color: transparent;

border-style: solid;

}

.popover > .arrow {

border-width: 11px;

}

.popover > .arrow:after {

content: "";

border-width: 10px;

}

.popover.top > .arrow {

bottom: -11px;

left: 50%;

margin-left: -11px;

border-top-color: #999;

border-top-color: rgba(0, 0, 0, .25);

border-bottom-width: 0;

}

.popover.top > .arrow:after {

bottom: 1px;

margin-left: -10px;

content: " ";

border-top-color: #fff;

border-bottom-width: 0;

}

.popover.right > .arrow {

top: 50%;

left: -11px;

margin-top: -11px;

border-right-color: #999;

border-right-color: rgba(0, 0, 0, .25);

border-left-width: 0;

}

.popover.right > .arrow:after {

bottom: -10px;

left: 1px;

content: " ";

border-right-color: #fff;

border-left-width: 0;

}

.popover.bottom > .arrow {

top: -11px;

left: 50%;

margin-left: -11px;

border-top-width: 0;

border-bottom-color: #999;

border-bottom-color: rgba(0, 0, 0, .25);

}

.popover.bottom > .arrow:after {

top: 1px;

margin-left: -10px;

content: " ";

border-top-width: 0;

border-bottom-color: #fff;

}

.popover.left > .arrow {

top: 50%;

right: -11px;

margin-top: -11px;

border-right-width: 0;

border-left-color: #999;

border-left-color: rgba(0, 0, 0, .25);

}

.popover.left > .arrow:after {

right: 1px;

bottom: -10px;

content: " ";

border-right-width: 0;

border-left-color: #fff;

}

.carousel {

position: relative;

}

.carousel-inner {

position: relative;

width: 100%;

overflow: hidden;

}

.carousel-inner > .item {

position: relative;

display: none;

-webkit-transition: .6s ease-in-out left;

-o-transition: .6s ease-in-out left;

transition: .6s ease-in-out left;

}

.carousel-inner > .item > img,

.carousel-inner > .item > a > img {

line-height: 1;

}

@media all and (transform-3d), (-webkit-transform-3d) {

.carousel-inner > .item {

-webkit-transition: -webkit-transform .6s ease-in-out;

-o-transition: -o-transform .6s ease-in-out;

transition: transform .6s ease-in-out;

-webkit-backface-visibility: hidden;

backface-visibility: hidden;

-webkit-perspective: 1000px;

perspective: 1000px;

}

.carousel-inner > .item.next,

.carousel-inner > .item.active.right {

left: 0;

-webkit-transform: translate3d(100%, 0, 0);

transform: translate3d(100%, 0, 0);

}

.carousel-inner > .item.prev,

.carousel-inner > .item.active.left {

left: 0;

-webkit-transform: translate3d(-100%, 0, 0);

transform: translate3d(-100%, 0, 0);

}

.carousel-inner > .item.next.left,

.carousel-inner > .item.prev.right,

.carousel-inner > .item.active {

left: 0;

-webkit-transform: translate3d(0, 0, 0);

transform: translate3d(0, 0, 0);

}

}

.carousel-inner > .active,

.carousel-inner > .next,

.carousel-inner > .prev {

display: block;

}

.carousel-inner > .active {

left: 0;

}

.carousel-inner > .next,

.carousel-inner > .prev {

position: absolute;

top: 0;

width: 100%;

}

.carousel-inner > .next {

left: 100%;

}

.carousel-inner > .prev {

left: -100%;

}

.carousel-inner > .next.left,

.carousel-inner > .prev.right {

left: 0;

}

.carousel-inner > .active.left {

left: -100%;

}

.carousel-inner > .active.right {

left: 100%;

}

.carousel-control {

position: absolute;

top: 0;

bottom: 0;

left: 0;

width: 15%;

font-size: 20px;

color: #fff;

text-align: center;

text-shadow: 0 1px 2px rgba(0, 0, 0, .6);

background-color: rgba(0, 0, 0, 0);

filter: alpha(opacity=50);

opacity: .5;

}

.carousel-control.left {

background-image: -webkit-linear-gradient(left, rgba(0, 0, 0, .5) 0%, rgba(0, 0, 0, .0001) 100%);

background-image: -o-linear-gradient(left, rgba(0, 0, 0, .5) 0%, rgba(0, 0, 0, .0001) 100%);

background-image: -webkit-gradient(linear, left top, right top, from(rgba(0, 0, 0, .5)), to(rgba(0, 0, 0, .0001)));

background-image: linear-gradient(to right, rgba(0, 0, 0, .5) 0%, rgba(0, 0, 0, .0001) 100%);

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#80000000', endColorstr='#00000000', GradientType=1);

background-repeat: repeat-x;

}

.carousel-control.right {

right: 0;

left: auto;

background-image: -webkit-linear-gradient(left, rgba(0, 0, 0, .0001) 0%, rgba(0, 0, 0, .5) 100%);

background-image: -o-linear-gradient(left, rgba(0, 0, 0, .0001) 0%, rgba(0, 0, 0, .5) 100%);

background-image: -webkit-gradient(linear, left top, right top, from(rgba(0, 0, 0, .0001)), to(rgba(0, 0, 0, .5)));

background-image: linear-gradient(to right, rgba(0, 0, 0, .0001) 0%, rgba(0, 0, 0, .5) 100%);

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#00000000', endColorstr='#80000000', GradientType=1);

background-repeat: repeat-x;

}

.carousel-control:hover,

.carousel-control:focus {

color: #fff;

text-decoration: none;

filter: alpha(opacity=90);

outline: 0;

opacity: .9;

}

.carousel-control .icon-prev,

.carousel-control .icon-next,

.carousel-control .glyphicon-chevron-left,

.carousel-control .glyphicon-chevron-right {

position: absolute;

top: 50%;

z-index: 5;

display: inline-block;

margin-top: -10px;

}

.carousel-control .icon-prev,

.carousel-control .glyphicon-chevron-left {

left: 50%;

margin-left: -10px;

}

.carousel-control .icon-next,

.carousel-control .glyphicon-chevron-right {

right: 50%;

margin-right: -10px;

}

.carousel-control .icon-prev,

.carousel-control .icon-next {

width: 20px;

height: 20px;

font-family: serif;

line-height: 1;

}

.carousel-control .icon-prev:before {

content: '\2039';

}

.carousel-control .icon-next:before {

content: '\203a';

}

.carousel-indicators {

position: absolute;

bottom: 10px;

left: 50%;

z-index: 15;

width: 60%;

padding-left: 0;

margin-left: -30%;

text-align: center;

list-style: none;

}

.carousel-indicators li {

display: inline-block;

width: 10px;

height: 10px;

margin: 1px;

text-indent: -999px;

cursor: pointer;

background-color: #000 \9;

background-color: rgba(0, 0, 0, 0);

border: 1px solid #fff;

border-radius: 10px;

}

.carousel-indicators .active {

width: 12px;

height: 12px;

margin: 0;

background-color: #fff;

}

.carousel-caption {

position: absolute;

right: 15%;

bottom: 20px;

left: 15%;

z-index: 10;

padding-top: 20px;

padding-bottom: 20px;

color: #fff;

text-align: center;

text-shadow: 0 1px 2px rgba(0, 0, 0, .6);

}

.carousel-caption .btn {

text-shadow: none;

}

@media screen and (min-width: 768px) {

.carousel-control .glyphicon-chevron-left,

.carousel-control .glyphicon-chevron-right,

.carousel-control .icon-prev,

.carousel-control .icon-next {

width: 30px;

height: 30px;

margin-top: -10px;

font-size: 30px;

}

.carousel-control .glyphicon-chevron-left,

.carousel-control .icon-prev {

margin-left: -10px;

}

.carousel-control .glyphicon-chevron-right,

.carousel-control .icon-next {

margin-right: -10px;

}

.carousel-caption {

right: 20%;

left: 20%;

padding-bottom: 30px;

}

.carousel-indicators {

bottom: 20px;

}

}

.clearfix:before,

.clearfix:after,

.dl-horizontal dd:before,

.dl-horizontal dd:after,

.container:before,

.container:after,

.container-fluid:before,

.container-fluid:after,

.row:before,

.row:after,

.form-horizontal .form-group:before,

.form-horizontal .form-group:after,

.btn-toolbar:before,

.btn-toolbar:after,

.btn-group-vertical > .btn-group:before,

.btn-group-vertical > .btn-group:after,

.nav:before,

.nav:after,

.navbar:before,

.navbar:after,

.navbar-header:before,

.navbar-header:after,

.navbar-collapse:before,

.navbar-collapse:after,

.pager:before,

.pager:after,

.panel-body:before,

.panel-body:after,

.modal-header:before,

.modal-header:after,

.modal-footer:before,

.modal-footer:after {

display: table;

content: " ";

}

.clearfix:after,

.dl-horizontal dd:after,

.container:after,

.container-fluid:after,

.row:after,

.form-horizontal .form-group:after,

.btn-toolbar:after,

.btn-group-vertical > .btn-group:after,

.nav:after,

.navbar:after,

.navbar-header:after,

.navbar-collapse:after,

.pager:after,

.panel-body:after,

.modal-header:after,

.modal-footer:after {

clear: both;

}

.center-block {

display: block;

margin-right: auto;

margin-left: auto;

}

.pull-right {

float: right !important;

}

.pull-left {

float: left !important;

}

.hide {

display: none !important;

}

.show {

display: block !important;

}

.invisible {

visibility: hidden;

}

.text-hide {

font: 0/0 a;

color: transparent;

text-shadow: none;

background-color: transparent;

border: 0;

}

.hidden {

display: none !important;

}

.affix {

position: fixed;

}

@-ms-viewport {

width: device-width;

}

.visible-xs,

.visible-sm,

.visible-md,

.visible-lg {

display: none !important;

}

.visible-xs-block,

.visible-xs-inline,

.visible-xs-inline-block,

.visible-sm-block,

.visible-sm-inline,

.visible-sm-inline-block,

.visible-md-block,

.visible-md-inline,

.visible-md-inline-block,

.visible-lg-block,

.visible-lg-inline,

.visible-lg-inline-block {

display: none !important;

}

@media (max-width: 767px) {

.visible-xs {

display: block !important;

}

table.visible-xs {

display: table !important;

}

tr.visible-xs {

display: table-row !important;

}

th.visible-xs,

td.visible-xs {

display: table-cell !important;

}

}

@media (max-width: 767px) {

.visible-xs-block {

display: block !important;

}

}

@media (max-width: 767px) {

.visible-xs-inline {

display: inline !important;

}

}

@media (max-width: 767px) {

.visible-xs-inline-block {

display: inline-block !important;

}

}

@media (min-width: 768px) and (max-width: 991px) {

.visible-sm {

display: block !important;

}

table.visible-sm {

display: table !important;

}

tr.visible-sm {

display: table-row !important;

}

th.visible-sm,

td.visible-sm {

display: table-cell !important;

}

}

@media (min-width: 768px) and (max-width: 991px) {

.visible-sm-block {

display: block !important;

}

}

@media (min-width: 768px) and (max-width: 991px) {

.visible-sm-inline {

display: inline !important;

}

}

@media (min-width: 768px) and (max-width: 991px) {

.visible-sm-inline-block {

display: inline-block !important;

}

}

@media (min-width: 992px) and (max-width: 1199px) {

.visible-md {

display: block !important;

}

table.visible-md {

display: table !important;

}

tr.visible-md {

display: table-row !important;

}

th.visible-md,

td.visible-md {

display: table-cell !important;

}

}

@media (min-width: 992px) and (max-width: 1199px) {

.visible-md-block {

display: block !important;

}

}

@media (min-width: 992px) and (max-width: 1199px) {

.visible-md-inline {

display: inline !important;

}

}

@media (min-width: 992px) and (max-width: 1199px) {

.visible-md-inline-block {

display: inline-block !important;

}

}

@media (min-width: 1200px) {

.visible-lg {

display: block !important;

}

table.visible-lg {

display: table !important;

}

tr.visible-lg {

display: table-row !important;

}

th.visible-lg,

td.visible-lg {

display: table-cell !important;

}

}

@media (min-width: 1200px) {

.visible-lg-block {

display: block !important;

}

}

@media (min-width: 1200px) {

.visible-lg-inline {

display: inline !important;

}

}

@media (min-width: 1200px) {

.visible-lg-inline-block {

display: inline-block !important;

}

}

@media (max-width: 767px) {

.hidden-xs {

display: none !important;

}

}

@media (min-width: 768px) and (max-width: 991px) {

.hidden-sm {

display: none !important;

}

}

@media (min-width: 992px) and (max-width: 1199px) {

.hidden-md {

display: none !important;

}

}

@media (min-width: 1200px) {

.hidden-lg {

display: none !important;

}

}

.visible-print {

display: none !important;

}

@media print {

.visible-print {

display: block !important;

}

table.visible-print {

display: table !important;

}

tr.visible-print {

display: table-row !important;

}

th.visible-print,

td.visible-print {

display: table-cell !important;

}

}

.visible-print-block {

display: none !important;

}

@media print {

.visible-print-block {

display: block !important;

}

}

.visible-print-inline {

display: none !important;

}

@media print {

.visible-print-inline {

display: inline !important;

}

}

.visible-print-inline-block {

display: none !important;

}

@media print {

.visible-print-inline-block {

display: inline-block !important;

}

}

@media print {

.hidden-print {

display: none !important;

}

}

/\*# sourceMappingURL=bootstrap.css.map \*/

/\*!

\* Datetimepicker for Bootstrap

\*

\* Copyright 2012 Stefan Petre

\* Improvements by Andrew Rowls

\* Licensed under the Apache License v2.0

\* <http://www.apache.org/licenses/LICENSE-2.0>

\*

\*/

.datetimepicker {

padding: 4px;

margin-top: 1px;

-webkit-border-radius: 4px;

-moz-border-radius: 4px;

border-radius: 4px;

direction: ltr;

}

.datetimepicker-inline {

width: 220px;

}

.datetimepicker.datetimepicker-rtl {

direction: rtl;

}

.datetimepicker.datetimepicker-rtl table tr td span {

float: right;

}

.datetimepicker-dropdown, .datetimepicker-dropdown-left {

top: 0;

left: 0;

}

[class\*=" datetimepicker-dropdown"]:before {

content: '';

display: inline-block;

border-left: 7px solid transparent;

border-right: 7px solid transparent;

border-bottom: 7px solid #cccccc;

border-bottom-color: rgba(0, 0, 0, 0.2);

position: absolute;

}

[class\*=" datetimepicker-dropdown"]:after {

content: '';

display: inline-block;

border-left: 6px solid transparent;

border-right: 6px solid transparent;

border-bottom: 6px solid #ffffff;

position: absolute;

}

[class\*=" datetimepicker-dropdown-top"]:before {

content: '';

display: inline-block;

border-left: 7px solid transparent;

border-right: 7px solid transparent;

border-top: 7px solid #cccccc;

border-top-color: rgba(0, 0, 0, 0.2);

border-bottom: 0;

}

[class\*=" datetimepicker-dropdown-top"]:after {

content: '';

display: inline-block;

border-left: 6px solid transparent;

border-right: 6px solid transparent;

border-top: 6px solid #ffffff;

border-bottom: 0;

}

.datetimepicker-dropdown-bottom-left:before {

top: -7px;

right: 6px;

}

.datetimepicker-dropdown-bottom-left:after {

top: -6px;

right: 7px;

}

.datetimepicker-dropdown-bottom-right:before {

top: -7px;

left: 6px;

}

.datetimepicker-dropdown-bottom-right:after {

top: -6px;

left: 7px;

}

.datetimepicker-dropdown-top-left:before {

bottom: -7px;

right: 6px;

}

.datetimepicker-dropdown-top-left:after {

bottom: -6px;

right: 7px;

}

.datetimepicker-dropdown-top-right:before {

bottom: -7px;

left: 6px;

}

.datetimepicker-dropdown-top-right:after {

bottom: -6px;

left: 7px;

}

.datetimepicker > div {

display: none;

}

.datetimepicker.minutes div.datetimepicker-minutes {

display: block;

}

.datetimepicker.hours div.datetimepicker-hours {

display: block;

}

.datetimepicker.days div.datetimepicker-days {

display: block;

}

.datetimepicker.months div.datetimepicker-months {

display: block;

}

.datetimepicker.years div.datetimepicker-years {

display: block;

}

.datetimepicker table {

margin: 0;

}

.datetimepicker td,

.datetimepicker th {

text-align: center;

width: 20px;

height: 20px;

-webkit-border-radius: 4px;

-moz-border-radius: 4px;

border-radius: 4px;

border: none;

}

.table-striped .datetimepicker table tr td,

.table-striped .datetimepicker table tr th {

background-color: transparent;

}

.datetimepicker table tr td.minute:hover {

background: #eeeeee;

cursor: pointer;

}

.datetimepicker table tr td.hour:hover {

background: #eeeeee;

cursor: pointer;

}

.datetimepicker table tr td.day:hover {

background: #eeeeee;

cursor: pointer;

}

.datetimepicker table tr td.old,

.datetimepicker table tr td.new {

color: #999999;

}

.datetimepicker table tr td.disabled,

.datetimepicker table tr td.disabled:hover {

background: none;

color: #999999;

cursor: default;

}

.datetimepicker table tr td.today,

.datetimepicker table tr td.today:hover,

.datetimepicker table tr td.today.disabled,

.datetimepicker table tr td.today.disabled:hover {

background-color: #fde19a;

background-image: -moz-linear-gradient(top, #fdd49a, #fdf59a);

background-image: -ms-linear-gradient(top, #fdd49a, #fdf59a);

background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#fdd49a), to(#fdf59a));

background-image: -webkit-linear-gradient(top, #fdd49a, #fdf59a);

background-image: -o-linear-gradient(top, #fdd49a, #fdf59a);

background-image: linear-gradient(top, #fdd49a, #fdf59a);

background-repeat: repeat-x;

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#fdd49a', endColorstr='#fdf59a', GradientType=0);

border-color: #fdf59a #fdf59a #fbed50;

border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);

filter: progid:DXImageTransform.Microsoft.gradient(enabled=false);

}

.datetimepicker table tr td.today:hover,

.datetimepicker table tr td.today:hover:hover,

.datetimepicker table tr td.today.disabled:hover,

.datetimepicker table tr td.today.disabled:hover:hover,

.datetimepicker table tr td.today:active,

.datetimepicker table tr td.today:hover:active,

.datetimepicker table tr td.today.disabled:active,

.datetimepicker table tr td.today.disabled:hover:active,

.datetimepicker table tr td.today.active,

.datetimepicker table tr td.today:hover.active,

.datetimepicker table tr td.today.disabled.active,

.datetimepicker table tr td.today.disabled:hover.active,

.datetimepicker table tr td.today.disabled,

.datetimepicker table tr td.today:hover.disabled,

.datetimepicker table tr td.today.disabled.disabled,

.datetimepicker table tr td.today.disabled:hover.disabled,

.datetimepicker table tr td.today[disabled],

.datetimepicker table tr td.today:hover[disabled],

.datetimepicker table tr td.today.disabled[disabled],

.datetimepicker table tr td.today.disabled:hover[disabled] {

background-color: #fdf59a;

}

.datetimepicker table tr td.today:active,

.datetimepicker table tr td.today:hover:active,

.datetimepicker table tr td.today.disabled:active,

.datetimepicker table tr td.today.disabled:hover:active,

.datetimepicker table tr td.today.active,

.datetimepicker table tr td.today:hover.active,

.datetimepicker table tr td.today.disabled.active,

.datetimepicker table tr td.today.disabled:hover.active {

background-color: #fbf069;

}

.datetimepicker table tr td.active,

.datetimepicker table tr td.active:hover,

.datetimepicker table tr td.active.disabled,

.datetimepicker table tr td.active.disabled:hover {

background-color: #006dcc;

background-image: -moz-linear-gradient(top, #0088cc, #0044cc);

background-image: -ms-linear-gradient(top, #0088cc, #0044cc);

background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#0088cc), to(#0044cc));

background-image: -webkit-linear-gradient(top, #0088cc, #0044cc);

background-image: -o-linear-gradient(top, #0088cc, #0044cc);

background-image: linear-gradient(top, #0088cc, #0044cc);

background-repeat: repeat-x;

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#0088cc', endColorstr='#0044cc', GradientType=0);

border-color: #0044cc #0044cc #002a80;

border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);

filter: progid:DXImageTransform.Microsoft.gradient(enabled=false);

color: #ffffff;

text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);

}

.datetimepicker table tr td.active:hover,

.datetimepicker table tr td.active:hover:hover,

.datetimepicker table tr td.active.disabled:hover,

.datetimepicker table tr td.active.disabled:hover:hover,

.datetimepicker table tr td.active:active,

.datetimepicker table tr td.active:hover:active,

.datetimepicker table tr td.active.disabled:active,

.datetimepicker table tr td.active.disabled:hover:active,

.datetimepicker table tr td.active.active,

.datetimepicker table tr td.active:hover.active,

.datetimepicker table tr td.active.disabled.active,

.datetimepicker table tr td.active.disabled:hover.active,

.datetimepicker table tr td.active.disabled,

.datetimepicker table tr td.active:hover.disabled,

.datetimepicker table tr td.active.disabled.disabled,

.datetimepicker table tr td.active.disabled:hover.disabled,

.datetimepicker table tr td.active[disabled],

.datetimepicker table tr td.active:hover[disabled],

.datetimepicker table tr td.active.disabled[disabled],

.datetimepicker table tr td.active.disabled:hover[disabled] {

background-color: #0044cc;

}

.datetimepicker table tr td.active:active,

.datetimepicker table tr td.active:hover:active,

.datetimepicker table tr td.active.disabled:active,

.datetimepicker table tr td.active.disabled:hover:active,

.datetimepicker table tr td.active.active,

.datetimepicker table tr td.active:hover.active,

.datetimepicker table tr td.active.disabled.active,

.datetimepicker table tr td.active.disabled:hover.active {

background-color: #003399;

}

.datetimepicker table tr td span {

display: block;

width: 23%;

height: 54px;

line-height: 54px;

float: left;

margin: 1%;

cursor: pointer;

-webkit-border-radius: 4px;

-moz-border-radius: 4px;

border-radius: 4px;

}

.datetimepicker .datetimepicker-hours span {

height: 26px;

line-height: 26px;

}

.datetimepicker .datetimepicker-hours table tr td span.hour\_am,

.datetimepicker .datetimepicker-hours table tr td span.hour\_pm {

width: 14.6%;

}

.datetimepicker .datetimepicker-hours fieldset legend,

.datetimepicker .datetimepicker-minutes fieldset legend {

margin-bottom: inherit;

line-height: 30px;

}

.datetimepicker .datetimepicker-minutes span {

height: 26px;

line-height: 26px;

}

.datetimepicker table tr td span:hover {

background: #eeeeee;

}

.datetimepicker table tr td span.disabled,

.datetimepicker table tr td span.disabled:hover {

background: none;

color: #999999;

cursor: default;

}

.datetimepicker table tr td span.active,

.datetimepicker table tr td span.active:hover,

.datetimepicker table tr td span.active.disabled,

.datetimepicker table tr td span.active.disabled:hover {

background-color: #006dcc;

background-image: -moz-linear-gradient(top, #0088cc, #0044cc);

background-image: -ms-linear-gradient(top, #0088cc, #0044cc);

background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#0088cc), to(#0044cc));

background-image: -webkit-linear-gradient(top, #0088cc, #0044cc);

background-image: -o-linear-gradient(top, #0088cc, #0044cc);

background-image: linear-gradient(top, #0088cc, #0044cc);

background-repeat: repeat-x;

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#0088cc', endColorstr='#0044cc', GradientType=0);

border-color: #0044cc #0044cc #002a80;

border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);

filter: progid:DXImageTransform.Microsoft.gradient(enabled=false);

color: #ffffff;

text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);

}

.datetimepicker table tr td span.active:hover,

.datetimepicker table tr td span.active:hover:hover,

.datetimepicker table tr td span.active.disabled:hover,

.datetimepicker table tr td span.active.disabled:hover:hover,

.datetimepicker table tr td span.active:active,

.datetimepicker table tr td span.active:hover:active,

.datetimepicker table tr td span.active.disabled:active,

.datetimepicker table tr td span.active.disabled:hover:active,

.datetimepicker table tr td span.active.active,

.datetimepicker table tr td span.active:hover.active,

.datetimepicker table tr td span.active.disabled.active,

.datetimepicker table tr td span.active.disabled:hover.active,

.datetimepicker table tr td span.active.disabled,

.datetimepicker table tr td span.active:hover.disabled,

.datetimepicker table tr td span.active.disabled.disabled,

.datetimepicker table tr td span.active.disabled:hover.disabled,

.datetimepicker table tr td span.active[disabled],

.datetimepicker table tr td span.active:hover[disabled],

.datetimepicker table tr td span.active.disabled[disabled],

.datetimepicker table tr td span.active.disabled:hover[disabled] {

background-color: #0044cc;

}

.datetimepicker table tr td span.active:active,

.datetimepicker table tr td span.active:hover:active,

.datetimepicker table tr td span.active.disabled:active,

.datetimepicker table tr td span.active.disabled:hover:active,

.datetimepicker table tr td span.active.active,

.datetimepicker table tr td span.active:hover.active,

.datetimepicker table tr td span.active.disabled.active,

.datetimepicker table tr td span.active.disabled:hover.active {

background-color: #003399;

}

.datetimepicker table tr td span.old {

color: #999999;

}

.datetimepicker th.switch {

width: 145px;

}

.datetimepicker th span.glyphicon {

pointer-events: none;

}

.datetimepicker thead tr:first-child th,

.datetimepicker tfoot th {

cursor: pointer;

}

.datetimepicker thead tr:first-child th:hover,

.datetimepicker tfoot th:hover {

background: #eeeeee;

}

.input-append.date .add-on i,

.input-prepend.date .add-on i,

.input-group.date .input-group-addon span {

cursor: pointer;

width: 14px;

height: 14px;

}

/\*

\* Table styles

\*/

table.dataTable {

width: 100%;

margin: 0 auto;

clear: both;

border-collapse: separate;

border-spacing: 0;

/\*

\* Header and footer styles

\*/

/\*

\* Body styles

\*/

}

table.dataTable thead th,

table.dataTable thead td,

table.dataTable tfoot th,

table.dataTable tfoot td {

padding: 4px 10px;

}

table.dataTable thead th,

table.dataTable tfoot th {

font-weight: bold;

}

table.dataTable thead th:active,

table.dataTable thead td:active {

outline: none;

}

table.dataTable thead .sorting\_asc,

table.dataTable thead .sorting\_desc,

table.dataTable thead .sorting {

cursor: pointer;

\*cursor: hand;

}

table.dataTable thead th div.DataTables\_sort\_wrapper {

position: relative;

padding-right: 10px;

}

table.dataTable thead th div.DataTables\_sort\_wrapper span {

position: absolute;

top: 50%;

margin-top: -8px;

right: -5px;

}

table.dataTable thead th.ui-state-default {

border-right-width: 0;

}

table.dataTable thead th.ui-state-default:last-child {

border-right-width: 1px;

}

table.dataTable tbody tr {

background-color: #ffffff;

}

table.dataTable tbody tr.selected {

background-color: #B0BED9;

}

table.dataTable tbody th,

table.dataTable tbody td {

padding: 8px 10px;

}

table.dataTable th.center,

table.dataTable td.center,

table.dataTable td.dataTables\_empty {

text-align: center;

}

table.dataTable th.right,

table.dataTable td.right {

text-align: right;

}

table.dataTable.row-border tbody th, table.dataTable.row-border tbody td, table.dataTable.display tbody th, table.dataTable.display tbody td {

border-top: 1px solid #ddd;

}

table.dataTable.row-border tbody tr:first-child th,

table.dataTable.row-border tbody tr:first-child td, table.dataTable.display tbody tr:first-child th,

table.dataTable.display tbody tr:first-child td {

border-top: none;

}

table.dataTable.cell-border tbody th, table.dataTable.cell-border tbody td {

border-top: 1px solid #ddd;

border-right: 1px solid #ddd;

}

table.dataTable.cell-border tbody tr th:first-child,

table.dataTable.cell-border tbody tr td:first-child {

border-left: 1px solid #ddd;

}

table.dataTable.cell-border tbody tr:first-child th,

table.dataTable.cell-border tbody tr:first-child td {

border-top: none;

}

table.dataTable.stripe tbody tr.odd, table.dataTable.display tbody tr.odd {

background-color: #f9f9f9;

}

table.dataTable.stripe tbody tr.odd.selected, table.dataTable.display tbody tr.odd.selected {

background-color: #abb9d3;

}

table.dataTable.hover tbody tr:hover,

table.dataTable.hover tbody tr.odd:hover,

table.dataTable.hover tbody tr.even:hover, table.dataTable.display tbody tr:hover,

table.dataTable.display tbody tr.odd:hover,

table.dataTable.display tbody tr.even:hover {

background-color: whitesmoke;

}

table.dataTable.hover tbody tr:hover.selected,

table.dataTable.hover tbody tr.odd:hover.selected,

table.dataTable.hover tbody tr.even:hover.selected, table.dataTable.display tbody tr:hover.selected,

table.dataTable.display tbody tr.odd:hover.selected,

table.dataTable.display tbody tr.even:hover.selected {

background-color: #a9b7d1;

}

table.dataTable.order-column tbody tr > .sorting\_1,

table.dataTable.order-column tbody tr > .sorting\_2,

table.dataTable.order-column tbody tr > .sorting\_3, table.dataTable.display tbody tr > .sorting\_1,

table.dataTable.display tbody tr > .sorting\_2,

table.dataTable.display tbody tr > .sorting\_3 {

background-color: #f9f9f9;

}

table.dataTable.order-column tbody tr.selected > .sorting\_1,

table.dataTable.order-column tbody tr.selected > .sorting\_2,

table.dataTable.order-column tbody tr.selected > .sorting\_3, table.dataTable.display tbody tr.selected > .sorting\_1,

table.dataTable.display tbody tr.selected > .sorting\_2,

table.dataTable.display tbody tr.selected > .sorting\_3 {

background-color: #acbad4;

}

table.dataTable.display tbody tr.odd > .sorting\_1, table.dataTable.order-column.stripe tbody tr.odd > .sorting\_1 {

background-color: #f1f1f1;

}

table.dataTable.display tbody tr.odd > .sorting\_2, table.dataTable.order-column.stripe tbody tr.odd > .sorting\_2 {

background-color: #f3f3f3;

}

table.dataTable.display tbody tr.odd > .sorting\_3, table.dataTable.order-column.stripe tbody tr.odd > .sorting\_3 {

background-color: whitesmoke;

}

table.dataTable.display tbody tr.odd.selected > .sorting\_1, table.dataTable.order-column.stripe tbody tr.odd.selected > .sorting\_1 {

background-color: #a6b3cd;

}

table.dataTable.display tbody tr.odd.selected > .sorting\_2, table.dataTable.order-column.stripe tbody tr.odd.selected > .sorting\_2 {

background-color: #a7b5ce;

}

table.dataTable.display tbody tr.odd.selected > .sorting\_3, table.dataTable.order-column.stripe tbody tr.odd.selected > .sorting\_3 {

background-color: #a9b6d0;

}

table.dataTable.display tbody tr.even > .sorting\_1, table.dataTable.order-column.stripe tbody tr.even > .sorting\_1 {

background-color: #f9f9f9;

}

table.dataTable.display tbody tr.even > .sorting\_2, table.dataTable.order-column.stripe tbody tr.even > .sorting\_2 {

background-color: #fbfbfb;

}

table.dataTable.display tbody tr.even > .sorting\_3, table.dataTable.order-column.stripe tbody tr.even > .sorting\_3 {

background-color: #fdfdfd;

}

table.dataTable.display tbody tr.even.selected > .sorting\_1, table.dataTable.order-column.stripe tbody tr.even.selected > .sorting\_1 {

background-color: #acbad4;

}

table.dataTable.display tbody tr.even.selected > .sorting\_2, table.dataTable.order-column.stripe tbody tr.even.selected > .sorting\_2 {

background-color: #adbbd6;

}

table.dataTable.display tbody tr.even.selected > .sorting\_3, table.dataTable.order-column.stripe tbody tr.even.selected > .sorting\_3 {

background-color: #afbdd8;

}

table.dataTable.display tbody tr:hover > .sorting\_1,

table.dataTable.display tbody tr.odd:hover > .sorting\_1,

table.dataTable.display tbody tr.even:hover > .sorting\_1, table.dataTable.order-column.hover tbody tr:hover > .sorting\_1,

table.dataTable.order-column.hover tbody tr.odd:hover > .sorting\_1,

table.dataTable.order-column.hover tbody tr.even:hover > .sorting\_1 {

background-color: #eaeaea;

}

table.dataTable.display tbody tr:hover > .sorting\_2,

table.dataTable.display tbody tr.odd:hover > .sorting\_2,

table.dataTable.display tbody tr.even:hover > .sorting\_2, table.dataTable.order-column.hover tbody tr:hover > .sorting\_2,

table.dataTable.order-column.hover tbody tr.odd:hover > .sorting\_2,

table.dataTable.order-column.hover tbody tr.even:hover > .sorting\_2 {

background-color: #ebebeb;

}

table.dataTable.display tbody tr:hover > .sorting\_3,

table.dataTable.display tbody tr.odd:hover > .sorting\_3,

table.dataTable.display tbody tr.even:hover > .sorting\_3, table.dataTable.order-column.hover tbody tr:hover > .sorting\_3,

table.dataTable.order-column.hover tbody tr.odd:hover > .sorting\_3,

table.dataTable.order-column.hover tbody tr.even:hover > .sorting\_3 {

background-color: #eeeeee;

}

table.dataTable.display tbody tr:hover.selected > .sorting\_1,

table.dataTable.display tbody tr.odd:hover.selected > .sorting\_1,

table.dataTable.display tbody tr.even:hover.selected > .sorting\_1, table.dataTable.order-column.hover tbody tr:hover.selected > .sorting\_1,

table.dataTable.order-column.hover tbody tr.odd:hover.selected > .sorting\_1,

table.dataTable.order-column.hover tbody tr.even:hover.selected > .sorting\_1 {

background-color: #a1aec7;

}

table.dataTable.display tbody tr:hover.selected > .sorting\_2,

table.dataTable.display tbody tr.odd:hover.selected > .sorting\_2,

table.dataTable.display tbody tr.even:hover.selected > .sorting\_2, table.dataTable.order-column.hover tbody tr:hover.selected > .sorting\_2,

table.dataTable.order-column.hover tbody tr.odd:hover.selected > .sorting\_2,

table.dataTable.order-column.hover tbody tr.even:hover.selected > .sorting\_2 {

background-color: #a2afc8;

}

table.dataTable.display tbody tr:hover.selected > .sorting\_3,

table.dataTable.display tbody tr.odd:hover.selected > .sorting\_3,

table.dataTable.display tbody tr.even:hover.selected > .sorting\_3, table.dataTable.order-column.hover tbody tr:hover.selected > .sorting\_3,

table.dataTable.order-column.hover tbody tr.odd:hover.selected > .sorting\_3,

table.dataTable.order-column.hover tbody tr.even:hover.selected > .sorting\_3 {

background-color: #a4b2cb;

}

table.dataTable.nowrap th, table.dataTable.nowrap td {

white-space: nowrap;

}

table.dataTable.compact thead th,

table.dataTable.compact thead td {

padding: 5px 9px;

}

table.dataTable.compact tfoot th,

table.dataTable.compact tfoot td {

padding: 5px 9px 3px 9px;

}

table.dataTable.compact tbody th,

table.dataTable.compact tbody td {

padding: 4px 5px;

}

table.dataTable th.dt-left,

table.dataTable td.dt-left {

text-align: left;

}

table.dataTable th.dt-center,

table.dataTable td.dt-center,

table.dataTable td.dataTables\_empty {

text-align: center;

}

table.dataTable th.dt-right,

table.dataTable td.dt-right {

text-align: right;

}

table.dataTable th.dt-justify,

table.dataTable td.dt-justify {

text-align: justify;

}

table.dataTable th.dt-nowrap,

table.dataTable td.dt-nowrap {

white-space: nowrap;

}

table.dataTable thead th.dt-head-left,

table.dataTable thead td.dt-head-left,

table.dataTable tfoot th.dt-head-left,

table.dataTable tfoot td.dt-head-left {

text-align: left;

}

table.dataTable thead th.dt-head-center,

table.dataTable thead td.dt-head-center,

table.dataTable tfoot th.dt-head-center,

table.dataTable tfoot td.dt-head-center {

text-align: center;

}

table.dataTable thead th.dt-head-right,

table.dataTable thead td.dt-head-right,

table.dataTable tfoot th.dt-head-right,

table.dataTable tfoot td.dt-head-right {

text-align: right;

}

table.dataTable thead th.dt-head-justify,

table.dataTable thead td.dt-head-justify,

table.dataTable tfoot th.dt-head-justify,

table.dataTable tfoot td.dt-head-justify {

text-align: justify;

}

table.dataTable thead th.dt-head-nowrap,

table.dataTable thead td.dt-head-nowrap,

table.dataTable tfoot th.dt-head-nowrap,

table.dataTable tfoot td.dt-head-nowrap {

white-space: nowrap;

}

table.dataTable tbody th.dt-body-left,

table.dataTable tbody td.dt-body-left {

text-align: left;

}

table.dataTable tbody th.dt-body-center,

table.dataTable tbody td.dt-body-center {

text-align: center;

}

table.dataTable tbody th.dt-body-right,

table.dataTable tbody td.dt-body-right {

text-align: right;

}

table.dataTable tbody th.dt-body-justify,

table.dataTable tbody td.dt-body-justify {

text-align: justify;

}

table.dataTable tbody th.dt-body-nowrap,

table.dataTable tbody td.dt-body-nowrap {

white-space: nowrap;

}

table.dataTable,

table.dataTable th,

table.dataTable td {

-webkit-box-sizing: content-box;

-moz-box-sizing: content-box;

box-sizing: content-box;

}

/\*

\* Control feature layout

\*/

.dataTables\_wrapper {

position: relative;

clear: both;

\*zoom: 1;

zoom: 1;

}

.dataTables\_wrapper .dataTables\_length {

float: left;

}

.dataTables\_wrapper .dataTables\_filter {

float: right;

text-align: right;

}

.dataTables\_wrapper .dataTables\_filter input {

margin-left: 0.5em;

}

.dataTables\_wrapper .dataTables\_info {

clear: both;

float: left;

padding-top: 0.55em;

}

.dataTables\_wrapper .dataTables\_paginate {

float: right;

text-align: right;

}

.dataTables\_wrapper .dataTables\_paginate .fg-button {

box-sizing: border-box;

display: inline-block;

min-width: 1.5em;

padding: 0.5em;

margin-left: 2px;

text-align: center;

text-decoration: none !important;

cursor: pointer;

\*cursor: hand;

color: #333 !important;

border: 1px solid transparent;

}

.dataTables\_wrapper .dataTables\_paginate .fg-button:active {

outline: none;

}

.dataTables\_wrapper .dataTables\_paginate .fg-button:first-child {

border-top-left-radius: 3px;

border-bottom-left-radius: 3px;

}

.dataTables\_wrapper .dataTables\_paginate .fg-button:last-child {

border-top-right-radius: 3px;

border-bottom-right-radius: 3px;

}

.dataTables\_wrapper .dataTables\_processing {

position: absolute;

top: 50%;

left: 50%;

width: 100%;

height: 40px;

margin-left: -50%;

margin-top: -25px;

padding-top: 20px;

text-align: center;

font-size: 1.2em;

background-color: white;

background: -webkit-gradient(linear, left top, right top, color-stop(0%, rgba(255, 255, 255, 0)), color-stop(25%, rgba(255, 255, 255, 0.9)), color-stop(75%, rgba(255, 255, 255, 0.9)), color-stop(100%, rgba(255, 255, 255, 0)));

/\* Chrome,Safari4+ \*/

background: -webkit-linear-gradient(left, rgba(255, 255, 255, 0) 0%, rgba(255, 255, 255, 0.9) 25%, rgba(255, 255, 255, 0.9) 75%, rgba(255, 255, 255, 0) 100%);

/\* Chrome10+,Safari5.1+ \*/

background: -moz-linear-gradient(left, rgba(255, 255, 255, 0) 0%, rgba(255, 255, 255, 0.9) 25%, rgba(255, 255, 255, 0.9) 75%, rgba(255, 255, 255, 0) 100%);

/\* FF3.6+ \*/

background: -ms-linear-gradient(left, rgba(255, 255, 255, 0) 0%, rgba(255, 255, 255, 0.9) 25%, rgba(255, 255, 255, 0.9) 75%, rgba(255, 255, 255, 0) 100%);

/\* IE10+ \*/

background: -o-linear-gradient(left, rgba(255, 255, 255, 0) 0%, rgba(255, 255, 255, 0.9) 25%, rgba(255, 255, 255, 0.9) 75%, rgba(255, 255, 255, 0) 100%);

/\* Opera 11.10+ \*/

background: linear-gradient(to right, rgba(255, 255, 255, 0) 0%, rgba(255, 255, 255, 0.9) 25%, rgba(255, 255, 255, 0.9) 75%, rgba(255, 255, 255, 0) 100%);

/\* W3C \*/

}

.dataTables\_wrapper .dataTables\_length,

.dataTables\_wrapper .dataTables\_filter,

.dataTables\_wrapper .dataTables\_info,

.dataTables\_wrapper .dataTables\_processing,

.dataTables\_wrapper .dataTables\_paginate {

color: #333;

}

.dataTables\_wrapper .dataTables\_scroll {

clear: both;

}

.dataTables\_wrapper .dataTables\_scrollBody {

\*margin-top: -1px;

-webkit-overflow-scrolling: touch;

}

.dataTables\_wrapper .ui-widget-header {

font-weight: normal;

}

.dataTables\_wrapper .ui-toolbar {

padding: 8px;

}

.dataTables\_wrapper:after {

visibility: hidden;

display: block;

content: "";

clear: both;

height: 0;

}

@media screen and (max-width: 767px) {

.dataTables\_wrapper .dataTables\_length,

.dataTables\_wrapper .dataTables\_filter,

.dataTables\_wrapper .dataTables\_info,

.dataTables\_wrapper .dataTables\_paginate {

float: none;

text-align: center;

}

.dataTables\_wrapper .dataTables\_filter,

.dataTables\_wrapper .dataTables\_paginate {

margin-top: 0.5em;

}

}

body {

background-color: transparent;

}

html {

background: url("/images/bg.png") no-repeat center center fixed;

-webkit-background-size: cover;

-moz-background-size: cover;

-o-background-size: cover;

background-size: cover;

}

img {

display: block;

margin-left: auto;

margin-right: auto;

}

.form-basic {

max-width: 640px;

margin: 0 auto;

padding: 55px;

box-sizing: border-box;

background-color: #ffffff;

box-shadow: 0 1px 3px 0 rgba(0, 0, 0, 0.1);

font: bold 14px sans-serif;

text-align: center;

}

.form-basic .form-row {

text-align: left;

margin-bottom: 22px;

}

.form-basic .form-title-row {

text-align: center;

margin-bottom: 55px;

}

/\* The form title \*/

.form-basic h1 {

display: inline-block;

box-sizing: border-box;

color: #4c565e;

font-size: 24px;

padding: 0 10px 15px;

border-bottom: 2px solid #6caee0;

margin: 0;

}

.form-basic .form-row > label span {

display: inline-block;

box-sizing: border-box;

color: #5F5F5F;

width: 180px;

text-align: right;

vertical-align: top;

padding: 12px 25px;

}

.form-basic input {

color: #5f5f5f;

box-sizing: border-box;

width: 240px;

box-shadow: 1px 2px 4px 0 rgba(0, 0, 0, 0.08);

padding: 12px;

border: 1px solid #dbdbdb;

}

.form-basic input[type=radio], .form-basic input[type=checkbox] {

box-shadow: none;

width: auto;

}

.form-basic input[type=checkbox] {

margin-top: 13px;

}

.form-basic select {

background-color: #ffffff;

color: #5f5f5f;

box-sizing: border-box;

max-width: 240px;

box-shadow: 1px 2px 4px 0 rgba(0, 0, 0, 0.08);

padding: 12px 8px;

border: 1px solid #dbdbdb;

}

.form-basic textarea {

color: #5f5f5f;

box-sizing: border-box;

width: 240px;

height: 80px;

box-shadow: 1px 2px 4px 0 rgba(0, 0, 0, 0.08);

font: normal 13px sans-serif;

padding: 12px;

border: 1px solid #dbdbdb;

resize: vertical;

}

.form-basic .form-radio-buttons {

display: inline-block;

vertical-align: top;

}

.form-basic .form-radio-buttons > div {

margin-top: 10px;

}

.form-basic .form-radio-buttons label span {

margin-left: 8px;

color: #5f5f5f;

font-weight: normal;

}

.form-basic .form-radio-buttons input {

width: auto;

}

.form-basic button {

display: block;

border-radius: 2px;

background-color: #6caee0;

color: #ffffff;

font-weight: bold;

box-shadow: 1px 2px 4px 0 rgba(0, 0, 0, 0.08);

padding: 14px 22px;

border: 0;

margin: 40px 183px 0;

}

/\* Making the form responsive. Remove this media query

if you don't need the form to work on mobile devices. \*/

@media ( max-width: 600px) {

.form-basic {

padding: 30px;

max-width: 480px;

}

.form-basic .form-row {

max-width: 300px;

margin: 25px auto;

text-align: left;

}

.form-basic .form-title-row {

margin-bottom: 50px;

}

.form-basic .form-row > label span {

display: block;

text-align: left;

padding: 0 0 15px;

}

.form-basic select {

width: 240px;

}

.form-basic input[type=checkbox] {

margin-top: 0;

}

.form-basic .form-radio-buttons > div {

margin: 0 0 10px;

}

.form-basic button {

margin: 0;

}

}

.main {

margin-top: 70px;

}

.main-center {

margin-top: 30px;

margin: 0 auto;

max-width: 330px;

padding: 40px 40px;

}

.responstable {

margin: 1em 0;

width: 100%;

overflow: hidden;

background: #FFF;

color: #024457;

border-radius: 10px;

border: 1px solid #167F92;

}

.responstable tr {

border: 1px solid #D9E4E6;

}

.responstable tr:nth-child(odd) {

background-color: #EAF3F3;

}

.responstable th {

display: none;

border: 1px solid #FFF;

background-color: #167F92;

color: #FFF;

padding: 1em;

}

.responstable th:first-child {

display: table-cell;

text-align: center;

}

.responstable th:nth-child(2) {

display: table-cell;

}

.responstable th:nth-child(2) span {

display: none;

}

.responstable th:nth-child(2):after {

content: attr(data-th);

}

@media ( min-width: 480px) {

.responstable th:nth-child(2) span {

display: block;

}

.responstable th:nth-child(2):after {

display: none;

}

}

.responstable td {

display: block;

word-wrap: break-word;

max-width: 7em;

}

.responstable td:first-child {

display: table-cell;

text-align: center;

border-right: 1px solid #D9E4E6;

}

@media ( min-width: 480px) {

.responstable td {

border: 1px solid #D9E4E6;

}

}

.responstable th, .responstable td {

text-align: left;

margin: .5em 1em;

}

@media ( min-width: 480px) {

.responstable th, .responstable td {

display: table-cell;

padding: 1em;

}

}

body {

padding: 0 2em;

font-family: Arial, sans-serif;

color: #024457;

/\*background: #f2f2f2;\*/

}

h1 {

font-family: Verdana;

font-weight: normal;

/\*color: #024457;\*/

}

h1 span {

color: #167F92;

}

.box {

border: 2px solid red;

}

hr {

-moz-border-bottom-colors: none;

-moz-border-image: none;

-moz-border-left-colors: none;

-moz-border-right-colors: none;

-moz-border-top-colors: none;

border-color: #EEEEEE -moz-use-text-color #FFFFFF;

border-style: solid none;

border-width: 1px 0;

margin: 18px 0;

border: 0;

height: 1px;

background: #333;

background-image: linear-gradient(to right, #ccc, #333, #ccc);

}

/\*!

\* Bootstrap v3.3.7 ([http://getbootstrap.com](http://getbootstrap.com/))

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under the MIT license

\*/

if (typeof jQuery === 'undefined') {

throw new Error('Bootstrap\'s JavaScript requires jQuery')

}

+function ($) {

'use strict';

var version = $.fn.jquery.split(' ')[0].split('.');

if ((version[0] < 2 && version[1] < 9) || (version[0] == 1 && version[1] == 9 && version[2] < 1) || (version[0] > 3)) {

throw new Error('Bootstrap\'s JavaScript requires jQuery version 1.9.1 or higher, but lower than version 4')

}

}(jQuery);

/\* ========================================================================

\* Bootstrap: transition.js v3.3.7

\* <http://getbootstrap.com/javascript/#transitions>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// CSS TRANSITION SUPPORT (Shoutout: <http://www.modernizr.com/>)

// ============================================================

function transitionEnd() {

var el = document.createElement('bootstrap');

var transEndEventNames = {

WebkitTransition: 'webkitTransitionEnd',

MozTransition: 'transitionend',

OTransition: 'oTransitionEnd otransitionend',

transition: 'transitionend'

};

for (var name in transEndEventNames) {

if (el.style[name] !== undefined) {

return {end: transEndEventNames[name]}

}

}

return false // explicit for ie8 ( .\_.)

}

// <http://blog.alexmaccaw.com/css-transitions>

$.fn.emulateTransitionEnd = function (duration) {

var called = false;

var $el = this;

$(this).one('bsTransitionEnd', function () {

called = true

});

var callback = function () {

if (!called) $($el).trigger($.support.transition.end)

};

setTimeout(callback, duration);

return this

};

$(function () {

$.support.transition = transitionEnd();

if (!$.support.transition) return;

$.event.special.bsTransitionEnd = {

bindType: $.support.transition.end,

delegateType: $.support.transition.end,

handle: function (e) {

if ($(e.target).is(this)) return e.handleObj.handler.apply(this, arguments)

}

}

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: alert.js v3.3.7

\* <http://getbootstrap.com/javascript/#alerts>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// ALERT CLASS DEFINITION

// ======================

var dismiss = '[data-dismiss="alert"]';

var Alert = function (el) {

$(el).on('click', dismiss, this.close)

};

Alert.VERSION = '3.3.7';

Alert.TRANSITION\_DURATION = 150;

Alert.prototype.close = function (e) {

var $this = $(this);

var selector = $this.attr('data-target');

if (!selector) {

selector = $this.attr('href');

selector = selector && selector.replace(/.\*(?=#[^\s]\*$)/, '') // strip for ie7

}

var $parent = $(selector === '#' ? [] : selector);

if (e) e.preventDefault();

if (!$parent.length) {

$parent = $this.closest('.alert')

}

$parent.trigger(e = $.Event('close.bs.alert'));

if (e.isDefaultPrevented()) return;

$parent.removeClass('in');

function removeElement() {

// detach from parent, fire event then clean up data

$parent.detach().trigger('closed.bs.alert').remove()

}

$.support.transition && $parent.hasClass('fade') ?

$parent

.one('bsTransitionEnd', removeElement)

.emulateTransitionEnd(Alert.TRANSITION\_DURATION) :

removeElement()

};

// ALERT PLUGIN DEFINITION

// =======================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.alert');

if (!data) $this.data('bs.alert', (data = new Alert(this)));

if (typeof option == 'string') data[option].call($this)

})

}

var old = $.fn.alert;

$.fn.alert = Plugin;

$.fn.alert.Constructor = Alert;

// ALERT NO CONFLICT

// =================

$.fn.alert.noConflict = function () {

$.fn.alert = old;

return this

};

// ALERT DATA-API

// ==============

$(document).on('click.bs.alert.data-api', dismiss, Alert.prototype.close)

}(jQuery);

/\* ========================================================================

\* Bootstrap: button.js v3.3.7

\* <http://getbootstrap.com/javascript/#buttons>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// BUTTON PUBLIC CLASS DEFINITION

// ==============================

var Button = function (element, options) {

this.$element = $(element);

this.options = $.extend({}, Button.DEFAULTS, options);

this.isLoading = false

};

Button.VERSION = '3.3.7';

Button.DEFAULTS = {

loadingText: 'loading...'

};

Button.prototype.setState = function (state) {

var d = 'disabled';

var $el = this.$element;

var val = $el.is('input') ? 'val' : 'html';

var data = $el.data();

state += 'Text';

if (data.resetText == null) $el.data('resetText', $el[val]());

// push to event loop to allow forms to submit

setTimeout($.proxy(function () {

$el[val](data[state] == null ? this.options[state] : data[state]);

if (state == 'loadingText') {

this.isLoading = true;

$el.addClass(d).attr(d, d).prop(d, true)

} else if (this.isLoading) {

this.isLoading = false;

$el.removeClass(d).removeAttr(d).prop(d, false)

}

}, this), 0)

};

Button.prototype.toggle = function () {

var changed = true;

var $parent = this.$element.closest('[data-toggle="buttons"]');

if ($parent.length) {

var $input = this.$element.find('input');

if ($input.prop('type') == 'radio') {

if ($input.prop('checked')) changed = false;

$parent.find('.active').removeClass('active');

this.$element.addClass('active')

} else if ($input.prop('type') == 'checkbox') {

if (($input.prop('checked')) !== this.$element.hasClass('active')) changed = false;

this.$element.toggleClass('active')

}

$input.prop('checked', this.$element.hasClass('active'));

if (changed) $input.trigger('change')

} else {

this.$element.attr('aria-pressed', !this.$element.hasClass('active'));

this.$element.toggleClass('active')

}

};

// BUTTON PLUGIN DEFINITION

// ========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.button');

var options = typeof option == 'object' && option;

if (!data) $this.data('bs.button', (data = new Button(this, options)));

if (option == 'toggle') data.toggle();

else if (option) data.setState(option)

})

}

var old = $.fn.button;

$.fn.button = Plugin;

$.fn.button.Constructor = Button;

// BUTTON NO CONFLICT

// ==================

$.fn.button.noConflict = function () {

$.fn.button = old;

return this

};

// BUTTON DATA-API

// ===============

$(document)

.on('click.bs.button.data-api', '[data-toggle^="button"]', function (e) {

var $btn = $(e.target).closest('.btn');

Plugin.call($btn, 'toggle');

if (!($(e.target).is('input[type="radio"], input[type="checkbox"]'))) {

// Prevent double click on radios, and the double selections (so cancellation) on checkboxes

e.preventDefault();

// The target component still receive the focus

if ($btn.is('input,button')) $btn.trigger('focus');

else $btn.find('input:visible,button:visible').first().trigger('focus')

}

})

.on('focus.bs.button.data-api blur.bs.button.data-api', '[data-toggle^="button"]', function (e) {

$(e.target).closest('.btn').toggleClass('focus', /^focus(in)?$/.test(e.type))

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: carousel.js v3.3.7

\* <http://getbootstrap.com/javascript/#carousel>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// CAROUSEL CLASS DEFINITION

// =========================

var Carousel = function (element, options) {

this.$element = $(element);

this.$indicators = this.$element.find('.carousel-indicators');

this.options = options;

this.paused = null;

this.sliding = null;

this.interval = null;

this.$active = null;

this.$items = null;

this.options.keyboard && this.$element.on('keydown.bs.carousel', $.proxy(this.keydown, this));

this.options.pause == 'hover' && !('ontouchstart' in document.documentElement) && this.$element

.on('mouseenter.bs.carousel', $.proxy(this.pause, this))

.on('mouseleave.bs.carousel', $.proxy(this.cycle, this))

};

Carousel.VERSION = '3.3.7';

Carousel.TRANSITION\_DURATION = 600;

Carousel.DEFAULTS = {

interval: 5000,

pause: 'hover',

wrap: true,

keyboard: true

};

Carousel.prototype.keydown = function (e) {

if (/input|textarea/i.test(e.target.tagName)) return;

switch (e.which) {

case 37:

this.prev();

break;

case 39:

this.next();

break;

default:

return

}

e.preventDefault()

};

Carousel.prototype.cycle = function (e) {

e || (this.paused = false);

this.interval && clearInterval(this.interval);

this.options.interval

&& !this.paused

&& (this.interval = setInterval($.proxy(this.next, this), this.options.interval));

return this

};

Carousel.prototype.getItemIndex = function (item) {

this.$items = item.parent().children('.item');

return this.$items.index(item || this.$active)

};

Carousel.prototype.getItemForDirection = function (direction, active) {

var activeIndex = this.getItemIndex(active);

var willWrap = (direction == 'prev' && activeIndex === 0)

|| (direction == 'next' && activeIndex == (this.$items.length - 1));

if (willWrap && !this.options.wrap) return active;

var delta = direction == 'prev' ? -1 : 1;

var itemIndex = (activeIndex + delta) % this.$items.length;

return this.$items.eq(itemIndex)

};

Carousel.prototype.to = function (pos) {

var that = this;

var activeIndex = this.getItemIndex(this.$active = this.$element.find('.item.active'));

if (pos > (this.$items.length - 1) || pos < 0) return;

if (this.sliding) return this.$element.one('slid.bs.carousel', function () {

that.to(pos)

}); // yes, "slid"

if (activeIndex == pos) return this.pause().cycle();

return this.slide(pos > activeIndex ? 'next' : 'prev', this.$items.eq(pos))

};

Carousel.prototype.pause = function (e) {

e || (this.paused = true);

if (this.$element.find('.next, .prev').length && $.support.transition) {

this.$element.trigger($.support.transition.end);

this.cycle(true)

}

this.interval = clearInterval(this.interval);

return this

};

Carousel.prototype.next = function () {

if (this.sliding) return;

return this.slide('next')

};

Carousel.prototype.prev = function () {

if (this.sliding) return;

return this.slide('prev')

};

Carousel.prototype.slide = function (type, next) {

var $active = this.$element.find('.item.active');

var $next = next || this.getItemForDirection(type, $active);

var isCycling = this.interval;

var direction = type == 'next' ? 'left' : 'right';

var that = this;

if ($next.hasClass('active')) return (this.sliding = false);

var relatedTarget = $next[0];

var slideEvent = $.Event('slide.bs.carousel', {

relatedTarget: relatedTarget,

direction: direction

});

this.$element.trigger(slideEvent);

if (slideEvent.isDefaultPrevented()) return;

this.sliding = true;

isCycling && this.pause();

if (this.$indicators.length) {

this.$indicators.find('.active').removeClass('active');

var $nextIndicator = $(this.$indicators.children()[this.getItemIndex($next)]);

$nextIndicator && $nextIndicator.addClass('active')

}

var slidEvent = $.Event('slid.bs.carousel', {relatedTarget: relatedTarget, direction: direction}); // yes, "slid"

if ($.support.transition && this.$element.hasClass('slide')) {

$next.addClass(type);

$next[0].offsetWidth; // force reflow

$active.addClass(direction);

$next.addClass(direction);

$active

.one('bsTransitionEnd', function () {

$next.removeClass([type, direction].join(' ')).addClass('active');

$active.removeClass(['active', direction].join(' '));

that.sliding = false;

setTimeout(function () {

that.$element.trigger(slidEvent)

}, 0)

})

.emulateTransitionEnd(Carousel.TRANSITION\_DURATION)

} else {

$active.removeClass('active');

$next.addClass('active');

this.sliding = false;

this.$element.trigger(slidEvent)

}

isCycling && this.cycle();

return this

};

// CAROUSEL PLUGIN DEFINITION

// ==========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.carousel');

var options = $.extend({}, Carousel.DEFAULTS, $this.data(), typeof option == 'object' && option);

var action = typeof option == 'string' ? option : options.slide;

if (!data) $this.data('bs.carousel', (data = new Carousel(this, options)));

if (typeof option == 'number') data.to(option);

else if (action) data[action]();

else if (options.interval) data.pause().cycle()

})

}

var old = $.fn.carousel;

$.fn.carousel = Plugin;

$.fn.carousel.Constructor = Carousel;

// CAROUSEL NO CONFLICT

// ====================

$.fn.carousel.noConflict = function () {

$.fn.carousel = old;

return this

};

// CAROUSEL DATA-API

// =================

var clickHandler = function (e) {

var href;

var $this = $(this);

var $target = $($this.attr('data-target') || (href = $this.attr('href')) && href.replace(/.\*(?=#[^\s]+$)/, '')); // strip for ie7

if (!$target.hasClass('carousel')) return;

var options = $.extend({}, $target.data(), $this.data());

var slideIndex = $this.attr('data-slide-to');

if (slideIndex) options.interval = false;

Plugin.call($target, options);

if (slideIndex) {

$target.data('bs.carousel').to(slideIndex)

}

e.preventDefault()

};

$(document)

.on('click.bs.carousel.data-api', '[data-slide]', clickHandler)

.on('click.bs.carousel.data-api', '[data-slide-to]', clickHandler);

$(window).on('load', function () {

$('[data-ride="carousel"]').each(function () {

var $carousel = $(this);

Plugin.call($carousel, $carousel.data())

})

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: collapse.js v3.3.7

\* <http://getbootstrap.com/javascript/#collapse>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

/\* jshint latedef: false \*/

+function ($) {

'use strict';

// COLLAPSE PUBLIC CLASS DEFINITION

// ================================

var Collapse = function (element, options) {

this.$element = $(element);

this.options = $.extend({}, Collapse.DEFAULTS, options);

this.$trigger = $('[data-toggle="collapse"][href="#' + element.id + '"],' +

'[data-toggle="collapse"][data-target="#' + element.id + '"]');

this.transitioning = null;

if (this.options.parent) {

this.$parent = this.getParent()

} else {

this.addAriaAndCollapsedClass(this.$element, this.$trigger)

}

if (this.options.toggle) this.toggle()

};

Collapse.VERSION = '3.3.7';

Collapse.TRANSITION\_DURATION = 350;

Collapse.DEFAULTS = {

toggle: true

};

Collapse.prototype.dimension = function () {

var hasWidth = this.$element.hasClass('width');

return hasWidth ? 'width' : 'height'

};

Collapse.prototype.show = function () {

if (this.transitioning || this.$element.hasClass('in')) return;

var activesData;

var actives = this.$parent && this.$parent.children('.panel').children('.in, .collapsing');

if (actives && actives.length) {

activesData = actives.data('bs.collapse');

if (activesData && activesData.transitioning) return

}

var startEvent = $.Event('show.bs.collapse');

this.$element.trigger(startEvent);

if (startEvent.isDefaultPrevented()) return;

if (actives && actives.length) {

Plugin.call(actives, 'hide');

activesData || actives.data('bs.collapse', null)

}

var dimension = this.dimension();

this.$element

.removeClass('collapse')

.addClass('collapsing')[dimension](0)

.attr('aria-expanded', true);

this.$trigger

.removeClass('collapsed')

.attr('aria-expanded', true);

this.transitioning = 1;

var complete = function () {

this.$element

.removeClass('collapsing')

.addClass('collapse in')[dimension]('');

this.transitioning = 0;

this.$element

.trigger('shown.bs.collapse')

};

if (!$.support.transition) return complete.call(this);

var scrollSize = $.camelCase(['scroll', dimension].join('-'));

this.$element

.one('bsTransitionEnd', $.proxy(complete, this))

.emulateTransitionEnd(Collapse.TRANSITION\_DURATION)[dimension](this.$element[0][scrollSize])

};

Collapse.prototype.hide = function () {

if (this.transitioning || !this.$element.hasClass('in')) return;

var startEvent = $.Event('hide.bs.collapse');

this.$element.trigger(startEvent);

if (startEvent.isDefaultPrevented()) return;

var dimension = this.dimension();

this.$element[dimension](this.$element[dimension]())[0].offsetHeight;

this.$element

.addClass('collapsing')

.removeClass('collapse in')

.attr('aria-expanded', false);

this.$trigger

.addClass('collapsed')

.attr('aria-expanded', false);

this.transitioning = 1;

var complete = function () {

this.transitioning = 0;

this.$element

.removeClass('collapsing')

.addClass('collapse')

.trigger('hidden.bs.collapse')

};

if (!$.support.transition) return complete.call(this);

this.$element

[dimension](0)

.one('bsTransitionEnd', $.proxy(complete, this))

.emulateTransitionEnd(Collapse.TRANSITION\_DURATION)

};

Collapse.prototype.toggle = function () {

this[this.$element.hasClass('in') ? 'hide' : 'show']()

};

Collapse.prototype.getParent = function () {

return $(this.options.parent)

.find('[data-toggle="collapse"][data-parent="' + this.options.parent + '"]')

.each($.proxy(function (i, element) {

var $element = $(element);

this.addAriaAndCollapsedClass(getTargetFromTrigger($element), $element)

}, this))

.end()

};

Collapse.prototype.addAriaAndCollapsedClass = function ($element, $trigger) {

var isOpen = $element.hasClass('in');

$element.attr('aria-expanded', isOpen);

$trigger

.toggleClass('collapsed', !isOpen)

.attr('aria-expanded', isOpen)

};

function getTargetFromTrigger($trigger) {

var href;

var target = $trigger.attr('data-target')

|| (href = $trigger.attr('href')) && href.replace(/.\*(?=#[^\s]+$)/, ''); // strip for ie7

return $(target)

}

// COLLAPSE PLUGIN DEFINITION

// ==========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.collapse');

var options = $.extend({}, Collapse.DEFAULTS, $this.data(), typeof option == 'object' && option);

if (!data && options.toggle && /show|hide/.test(option)) options.toggle = false;

if (!data) $this.data('bs.collapse', (data = new Collapse(this, options)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.collapse;

$.fn.collapse = Plugin;

$.fn.collapse.Constructor = Collapse;

// COLLAPSE NO CONFLICT

// ====================

$.fn.collapse.noConflict = function () {

$.fn.collapse = old;

return this

};

// COLLAPSE DATA-API

// =================

$(document).on('click.bs.collapse.data-api', '[data-toggle="collapse"]', function (e) {

var $this = $(this);

if (!$this.attr('data-target')) e.preventDefault();

var $target = getTargetFromTrigger($this);

var data = $target.data('bs.collapse');

var option = data ? 'toggle' : $this.data();

Plugin.call($target, option)

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: dropdown.js v3.3.7

\* <http://getbootstrap.com/javascript/#dropdowns>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// DROPDOWN CLASS DEFINITION

// =========================

var backdrop = '.dropdown-backdrop';

var toggle = '[data-toggle="dropdown"]';

var Dropdown = function (element) {

$(element).on('click.bs.dropdown', this.toggle)

};

Dropdown.VERSION = '3.3.7';

function getParent($this) {

var selector = $this.attr('data-target');

if (!selector) {

selector = $this.attr('href');

selector = selector && /#[A-Za-z]/.test(selector) && selector.replace(/.\*(?=#[^\s]\*$)/, '') // strip for ie7

}

var $parent = selector && $(selector);

return $parent && $parent.length ? $parent : $this.parent()

}

function clearMenus(e) {

if (e && e.which === 3) return;

$(backdrop).remove();

$(toggle).each(function () {

var $this = $(this);

var $parent = getParent($this);

var relatedTarget = {relatedTarget: this};

if (!$parent.hasClass('open')) return;

if (e && e.type == 'click' && /input|textarea/i.test(e.target.tagName) && $.contains($parent[0], e.target)) return;

$parent.trigger(e = $.Event('hide.bs.dropdown', relatedTarget));

if (e.isDefaultPrevented()) return;

$this.attr('aria-expanded', 'false');

$parent.removeClass('open').trigger($.Event('hidden.bs.dropdown', relatedTarget))

})

}

Dropdown.prototype.toggle = function (e) {

var $this = $(this);

if ($this.is('.disabled, :disabled')) return;

var $parent = getParent($this);

var isActive = $parent.hasClass('open');

clearMenus();

if (!isActive) {

if ('ontouchstart' in document.documentElement && !$parent.closest('.navbar-nav').length) {

// if mobile we use a backdrop because click events don't delegate

$(document.createElement('div'))

.addClass('dropdown-backdrop')

.insertAfter($(this))

.on('click', clearMenus)

}

var relatedTarget = {relatedTarget: this};

$parent.trigger(e = $.Event('show.bs.dropdown', relatedTarget));

if (e.isDefaultPrevented()) return;

$this

.trigger('focus')

.attr('aria-expanded', 'true');

$parent

.toggleClass('open')

.trigger($.Event('shown.bs.dropdown', relatedTarget))

}

return false

};

Dropdown.prototype.keydown = function (e) {

if (!/(38|40|27|32)/.test(e.which) || /input|textarea/i.test(e.target.tagName)) return;

var $this = $(this);

e.preventDefault();

e.stopPropagation();

if ($this.is('.disabled, :disabled')) return;

var $parent = getParent($this);

var isActive = $parent.hasClass('open');

if (!isActive && e.which != 27 || isActive && e.which == 27) {

if (e.which == 27) $parent.find(toggle).trigger('focus');

return $this.trigger('click')

}

var desc = ' li:not(.disabled):visible a';

var $items = $parent.find('.dropdown-menu' + desc);

if (!$items.length) return;

var index = $items.index(e.target);

if (e.which == 38 && index > 0) index--; // up

if (e.which == 40 && index < $items.length - 1) index++; // down

if (!~index) index = 0;

$items.eq(index).trigger('focus')

};

// DROPDOWN PLUGIN DEFINITION

// ==========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.dropdown');

if (!data) $this.data('bs.dropdown', (data = new Dropdown(this)));

if (typeof option == 'string') data[option].call($this)

})

}

var old = $.fn.dropdown;

$.fn.dropdown = Plugin;

$.fn.dropdown.Constructor = Dropdown;

// DROPDOWN NO CONFLICT

// ====================

$.fn.dropdown.noConflict = function () {

$.fn.dropdown = old;

return this

};

// APPLY TO STANDARD DROPDOWN ELEMENTS

// ===================================

$(document)

.on('click.bs.dropdown.data-api', clearMenus)

.on('click.bs.dropdown.data-api', '.dropdown form', function (e) {

e.stopPropagation()

})

.on('click.bs.dropdown.data-api', toggle, Dropdown.prototype.toggle)

.on('keydown.bs.dropdown.data-api', toggle, Dropdown.prototype.keydown)

.on('keydown.bs.dropdown.data-api', '.dropdown-menu', Dropdown.prototype.keydown)

}(jQuery);

/\* ========================================================================

\* Bootstrap: modal.js v3.3.7

\* <http://getbootstrap.com/javascript/#modals>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// MODAL CLASS DEFINITION

// ======================

var Modal = function (element, options) {

this.options = options;

this.$body = $(document.body);

this.$element = $(element);

this.$dialog = this.$element.find('.modal-dialog');

this.$backdrop = null;

this.isShown = null;

this.originalBodyPad = null;

this.scrollbarWidth = 0;

this.ignoreBackdropClick = false;

if (this.options.remote) {

this.$element

.find('.modal-content')

.load(this.options.remote, $.proxy(function () {

this.$element.trigger('loaded.bs.modal')

}, this))

}

};

Modal.VERSION = '3.3.7';

Modal.TRANSITION\_DURATION = 300;

Modal.BACKDROP\_TRANSITION\_DURATION = 150;

Modal.DEFAULTS = {

backdrop: true,

keyboard: true,

show: true

};

Modal.prototype.toggle = function (\_relatedTarget) {

return this.isShown ? this.hide() : this.show(\_relatedTarget)

};

Modal.prototype.show = function (\_relatedTarget) {

var that = this;

var e = $.Event('show.bs.modal', {relatedTarget: \_relatedTarget});

this.$element.trigger(e);

if (this.isShown || e.isDefaultPrevented()) return;

this.isShown = true;

this.checkScrollbar();

this.setScrollbar();

this.$body.addClass('modal-open');

this.escape();

this.resize();

this.$element.on('click.dismiss.bs.modal', '[data-dismiss="modal"]', $.proxy(this.hide, this));

this.$dialog.on('mousedown.dismiss.bs.modal', function () {

that.$element.one('mouseup.dismiss.bs.modal', function (e) {

if ($(e.target).is(that.$element)) that.ignoreBackdropClick = true

})

});

this.backdrop(function () {

var transition = $.support.transition && that.$element.hasClass('fade');

if (!that.$element.parent().length) {

that.$element.appendTo(that.$body) // don't move modals dom position

}

that.$element

.show()

.scrollTop(0);

that.adjustDialog();

if (transition) {

that.$element[0].offsetWidth // force reflow

}

that.$element.addClass('in');

that.enforceFocus();

var e = $.Event('shown.bs.modal', {relatedTarget: \_relatedTarget});

transition ?

that.$dialog // wait for modal to slide in

.one('bsTransitionEnd', function () {

that.$element.trigger('focus').trigger(e)

})

.emulateTransitionEnd(Modal.TRANSITION\_DURATION) :

that.$element.trigger('focus').trigger(e)

})

};

Modal.prototype.hide = function (e) {

if (e) e.preventDefault();

e = $.Event('hide.bs.modal');

this.$element.trigger(e);

if (!this.isShown || e.isDefaultPrevented()) return;

this.isShown = false;

this.escape();

this.resize();

$(document).off('focusin.bs.modal');

this.$element

.removeClass('in')

.off('click.dismiss.bs.modal')

.off('mouseup.dismiss.bs.modal');

this.$dialog.off('mousedown.dismiss.bs.modal');

$.support.transition && this.$element.hasClass('fade') ?

this.$element

.one('bsTransitionEnd', $.proxy(this.hideModal, this))

.emulateTransitionEnd(Modal.TRANSITION\_DURATION) :

this.hideModal()

};

Modal.prototype.enforceFocus = function () {

$(document)

.off('focusin.bs.modal') // guard against infinite focus loop

.on('focusin.bs.modal', $.proxy(function (e) {

if (document !== e.target &&

this.$element[0] !== e.target &&

!this.$element.has(e.target).length) {

this.$element.trigger('focus')

}

}, this))

};

Modal.prototype.escape = function () {

if (this.isShown && this.options.keyboard) {

this.$element.on('keydown.dismiss.bs.modal', $.proxy(function (e) {

e.which == 27 && this.hide()

}, this))

} else if (!this.isShown) {

this.$element.off('keydown.dismiss.bs.modal')

}

};

Modal.prototype.resize = function () {

if (this.isShown) {

$(window).on('resize.bs.modal', $.proxy(this.handleUpdate, this))

} else {

$(window).off('resize.bs.modal')

}

};

Modal.prototype.hideModal = function () {

var that = this;

this.$element.hide();

this.backdrop(function () {

that.$body.removeClass('modal-open');

that.resetAdjustments();

that.resetScrollbar();

that.$element.trigger('hidden.bs.modal')

})

};

Modal.prototype.removeBackdrop = function () {

this.$backdrop && this.$backdrop.remove();

this.$backdrop = null

};

Modal.prototype.backdrop = function (callback) {

var that = this;

var animate = this.$element.hasClass('fade') ? 'fade' : '';

if (this.isShown && this.options.backdrop) {

var doAnimate = $.support.transition && animate;

this.$backdrop = $(document.createElement('div'))

.addClass('modal-backdrop ' + animate)

.appendTo(this.$body);

this.$element.on('click.dismiss.bs.modal', $.proxy(function (e) {

if (this.ignoreBackdropClick) {

this.ignoreBackdropClick = false;

return

}

if (e.target !== e.currentTarget) return;

this.options.backdrop == 'static'

? this.$element[0].focus()

: this.hide()

}, this));

if (doAnimate) this.$backdrop[0].offsetWidth; // force reflow

this.$backdrop.addClass('in');

if (!callback) return;

doAnimate ?

this.$backdrop

.one('bsTransitionEnd', callback)

.emulateTransitionEnd(Modal.BACKDROP\_TRANSITION\_DURATION) :

callback()

} else if (!this.isShown && this.$backdrop) {

this.$backdrop.removeClass('in');

var callbackRemove = function () {

that.removeBackdrop();

callback && callback()

};

$.support.transition && this.$element.hasClass('fade') ?

this.$backdrop

.one('bsTransitionEnd', callbackRemove)

.emulateTransitionEnd(Modal.BACKDROP\_TRANSITION\_DURATION) :

callbackRemove()

} else if (callback) {

callback()

}

};

// these following methods are used to handle overflowing modals

Modal.prototype.handleUpdate = function () {

this.adjustDialog()

};

Modal.prototype.adjustDialog = function () {

var modalIsOverflowing = this.$element[0].scrollHeight > document.documentElement.clientHeight;

this.$element.css({

paddingLeft: !this.bodyIsOverflowing && modalIsOverflowing ? this.scrollbarWidth : '',

paddingRight: this.bodyIsOverflowing && !modalIsOverflowing ? this.scrollbarWidth : ''

})

};

Modal.prototype.resetAdjustments = function () {

this.$element.css({

paddingLeft: '',

paddingRight: ''

})

};

Modal.prototype.checkScrollbar = function () {

var fullWindowWidth = window.innerWidth;

if (!fullWindowWidth) { // workaround for missing window.innerWidth in IE8

var documentElementRect = document.documentElement.getBoundingClientRect();

fullWindowWidth = documentElementRect.right - Math.abs(documentElementRect.left)

}

this.bodyIsOverflowing = document.body.clientWidth < fullWindowWidth;

this.scrollbarWidth = this.measureScrollbar()

};

Modal.prototype.setScrollbar = function () {

var bodyPad = parseInt((this.$body.css('padding-right') || 0), 10);

this.originalBodyPad = document.body.style.paddingRight || '';

if (this.bodyIsOverflowing) this.$body.css('padding-right', bodyPad + this.scrollbarWidth)

};

Modal.prototype.resetScrollbar = function () {

this.$body.css('padding-right', this.originalBodyPad)

};

Modal.prototype.measureScrollbar = function () { // thx walsh

var scrollDiv = document.createElement('div');

scrollDiv.className = 'modal-scrollbar-measure';

this.$body.append(scrollDiv);

var scrollbarWidth = scrollDiv.offsetWidth - scrollDiv.clientWidth;

this.$body[0].removeChild(scrollDiv);

return scrollbarWidth

};

// MODAL PLUGIN DEFINITION

// =======================

function Plugin(option, \_relatedTarget) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.modal');

var options = $.extend({}, Modal.DEFAULTS, $this.data(), typeof option == 'object' && option);

if (!data) $this.data('bs.modal', (data = new Modal(this, options)));

if (typeof option == 'string') data[option](\_relatedTarget);

else if (options.show) data.show(\_relatedTarget)

})

}

var old = $.fn.modal;

$.fn.modal = Plugin;

$.fn.modal.Constructor = Modal;

// MODAL NO CONFLICT

// =================

$.fn.modal.noConflict = function () {

$.fn.modal = old;

return this

};

// MODAL DATA-API

// ==============

$(document).on('click.bs.modal.data-api', '[data-toggle="modal"]', function (e) {

var $this = $(this);

var href = $this.attr('href');

var $target = $($this.attr('data-target') || (href && href.replace(/.\*(?=#[^\s]+$)/, ''))); // strip for ie7

var option = $target.data('bs.modal') ? 'toggle' : $.extend({remote: !/#/.test(href) && href}, $target.data(), $this.data());

if ($this.is('a')) e.preventDefault();

$target.one('show.bs.modal', function (showEvent) {

if (showEvent.isDefaultPrevented()) return; // only register focus restorer if modal will actually get shown

$target.one('hidden.bs.modal', function () {

$this.is(':visible') && $this.trigger('focus')

})

});

Plugin.call($target, option, this)

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: tooltip.js v3.3.7

\* <http://getbootstrap.com/javascript/#tooltip>

\* Inspired by the original jQuery.tipsy by Jason Frame

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// TOOLTIP PUBLIC CLASS DEFINITION

// ===============================

var Tooltip = function (element, options) {

this.type = null;

this.options = null;

this.enabled = null;

this.timeout = null;

this.hoverState = null;

this.$element = null;

this.inState = null;

this.init('tooltip', element, options)

};

Tooltip.VERSION = '3.3.7';

Tooltip.TRANSITION\_DURATION = 150;

Tooltip.DEFAULTS = {

animation: true,

placement: 'top',

selector: false,

template: '<div class="tooltip" role="tooltip"><div class="tooltip-arrow"></div><div class="tooltip-inner"></div></div>',

trigger: 'hover focus',

title: '',

delay: 0,

html: false,

container: false,

viewport: {

selector: 'body',

padding: 0

}

};

Tooltip.prototype.init = function (type, element, options) {

this.enabled = true;

this.type = type;

this.$element = $(element);

this.options = this.getOptions(options);

this.$viewport = this.options.viewport && $($.isFunction(this.options.viewport) ? this.options.viewport.call(this, this.$element) : (this.options.viewport.selector || this.options.viewport));

this.inState = {click: false, hover: false, focus: false};

if (this.$element[0] instanceof document.constructor && !this.options.selector) {

throw new Error('`selector` option must be specified when initializing ' + this.type + ' on the window.document object!')

}

var triggers = this.options.trigger.split(' ');

for (var i = triggers.length; i--;) {

var trigger = triggers[i];

if (trigger == 'click') {

this.$element.on('click.' + this.type, this.options.selector, $.proxy(this.toggle, this))

} else if (trigger != 'manual') {

var eventIn = trigger == 'hover' ? 'mouseenter' : 'focusin';

var eventOut = trigger == 'hover' ? 'mouseleave' : 'focusout';

this.$element.on(eventIn + '.' + this.type, this.options.selector, $.proxy(this.enter, this));

this.$element.on(eventOut + '.' + this.type, this.options.selector, $.proxy(this.leave, this))

}

}

this.options.selector ?

(this.\_options = $.extend({}, this.options, {trigger: 'manual', selector: ''})) :

this.fixTitle()

};

Tooltip.prototype.getDefaults = function () {

return Tooltip.DEFAULTS

};

Tooltip.prototype.getOptions = function (options) {

options = $.extend({}, this.getDefaults(), this.$element.data(), options);

if (options.delay && typeof options.delay == 'number') {

options.delay = {

show: options.delay,

hide: options.delay

}

}

return options

};

Tooltip.prototype.getDelegateOptions = function () {

var options = {};

var defaults = this.getDefaults();

this.\_options && $.each(this.\_options, function (key, value) {

if (defaults[key] != value) options[key] = value

});

return options

};

Tooltip.prototype.enter = function (obj) {

var self = obj instanceof this.constructor ?

obj : $(obj.currentTarget).data('bs.' + this.type);

if (!self) {

self = new this.constructor(obj.currentTarget, this.getDelegateOptions());

$(obj.currentTarget).data('bs.' + this.type, self)

}

if (obj instanceof $.Event) {

self.inState[obj.type == 'focusin' ? 'focus' : 'hover'] = true

}

if (self.tip().hasClass('in') || self.hoverState == 'in') {

self.hoverState = 'in';

return

}

clearTimeout(self.timeout);

self.hoverState = 'in';

if (!self.options.delay || !self.options.delay.show) return self.show();

self.timeout = setTimeout(function () {

if (self.hoverState == 'in') self.show()

}, self.options.delay.show)

};

Tooltip.prototype.isInStateTrue = function () {

for (var key in this.inState) {

if (this.inState[key]) return true

}

return false

};

Tooltip.prototype.leave = function (obj) {

var self = obj instanceof this.constructor ?

obj : $(obj.currentTarget).data('bs.' + this.type);

if (!self) {

self = new this.constructor(obj.currentTarget, this.getDelegateOptions());

$(obj.currentTarget).data('bs.' + this.type, self)

}

if (obj instanceof $.Event) {

self.inState[obj.type == 'focusout' ? 'focus' : 'hover'] = false

}

if (self.isInStateTrue()) return;

clearTimeout(self.timeout);

self.hoverState = 'out';

if (!self.options.delay || !self.options.delay.hide) return self.hide();

self.timeout = setTimeout(function () {

if (self.hoverState == 'out') self.hide()

}, self.options.delay.hide)

};

Tooltip.prototype.show = function () {

var e = $.Event('show.bs.' + this.type);

if (this.hasContent() && this.enabled) {

this.$element.trigger(e);

var inDom = $.contains(this.$element[0].ownerDocument.documentElement, this.$element[0]);

if (e.isDefaultPrevented() || !inDom) return;

var that = this;

var $tip = this.tip();

var tipId = this.getUID(this.type);

this.setContent();

$tip.attr('id', tipId);

this.$element.attr('aria-describedby', tipId);

if (this.options.animation) $tip.addClass('fade');

var placement = typeof this.options.placement == 'function' ?

this.options.placement.call(this, $tip[0], this.$element[0]) :

this.options.placement;

var autoToken = /\s?auto?\s?/i;

var autoPlace = autoToken.test(placement);

if (autoPlace) placement = placement.replace(autoToken, '') || 'top';

$tip

.detach()

.css({top: 0, left: 0, display: 'block'})

.addClass(placement)

.data('bs.' + this.type, this);

this.options.container ? $tip.appendTo(this.options.container) : $tip.insertAfter(this.$element);

this.$element.trigger('inserted.bs.' + this.type);

var pos = this.getPosition();

var actualWidth = $tip[0].offsetWidth;

var actualHeight = $tip[0].offsetHeight;

if (autoPlace) {

var orgPlacement = placement;

var viewportDim = this.getPosition(this.$viewport);

placement = placement == 'bottom' && pos.bottom + actualHeight > viewportDim.bottom ? 'top' :

placement == 'top' && pos.top - actualHeight < viewportDim.top ? 'bottom' :

placement == 'right' && pos.right + actualWidth > viewportDim.width ? 'left' :

placement == 'left' && pos.left - actualWidth < viewportDim.left ? 'right' :

placement;

$tip

.removeClass(orgPlacement)

.addClass(placement)

}

var calculatedOffset = this.getCalculatedOffset(placement, pos, actualWidth, actualHeight);

this.applyPlacement(calculatedOffset, placement);

var complete = function () {

var prevHoverState = that.hoverState;

that.$element.trigger('shown.bs.' + that.type);

that.hoverState = null;

if (prevHoverState == 'out') that.leave(that)

};

$.support.transition && this.$tip.hasClass('fade') ?

$tip

.one('bsTransitionEnd', complete)

.emulateTransitionEnd(Tooltip.TRANSITION\_DURATION) :

complete()

}

};

Tooltip.prototype.applyPlacement = function (offset, placement) {

var $tip = this.tip();

var width = $tip[0].offsetWidth;

var height = $tip[0].offsetHeight;

// manually read margins because getBoundingClientRect includes difference

var marginTop = parseInt($tip.css('margin-top'), 10);

var marginLeft = parseInt($tip.css('margin-left'), 10);

// we must check for NaN for ie 8/9

if (isNaN(marginTop)) marginTop = 0;

if (isNaN(marginLeft)) marginLeft = 0;

offset.top += marginTop;

offset.left += marginLeft;

// $.fn.offset doesn't round pixel values

// so we use setOffset directly with our own function B-0

$.offset.setOffset($tip[0], $.extend({

using: function (props) {

$tip.css({

top: Math.round(props.top),

left: Math.round(props.left)

})

}

}, offset), 0);

$tip.addClass('in');

// check to see if placing tip in new offset caused the tip to resize itself

var actualWidth = $tip[0].offsetWidth;

var actualHeight = $tip[0].offsetHeight;

if (placement == 'top' && actualHeight != height) {

offset.top = offset.top + height - actualHeight

}

var delta = this.getViewportAdjustedDelta(placement, offset, actualWidth, actualHeight);

if (delta.left) offset.left += delta.left;

else offset.top += delta.top;

var isVertical = /top|bottom/.test(placement);

var arrowDelta = isVertical ? delta.left \* 2 - width + actualWidth : delta.top \* 2 - height + actualHeight;

var arrowOffsetPosition = isVertical ? 'offsetWidth' : 'offsetHeight';

$tip.offset(offset);

this.replaceArrow(arrowDelta, $tip[0][arrowOffsetPosition], isVertical)

};

Tooltip.prototype.replaceArrow = function (delta, dimension, isVertical) {

this.arrow()

.css(isVertical ? 'left' : 'top', 50 \* (1 - delta / dimension) + '%')

.css(isVertical ? 'top' : 'left', '')

};

Tooltip.prototype.setContent = function () {

var $tip = this.tip();

var title = this.getTitle();

$tip.find('.tooltip-inner')[this.options.html ? 'html' : 'text'](title);

$tip.removeClass('fade in top bottom left right')

};

Tooltip.prototype.hide = function (callback) {

var that = this;

var $tip = $(this.$tip);

var e = $.Event('hide.bs.' + this.type);

function complete() {

if (that.hoverState != 'in') $tip.detach();

if (that.$element) { // TODO: Check whether guarding this code with this `if` is really necessary.

that.$element

.removeAttr('aria-describedby')

.trigger('hidden.bs.' + that.type)

}

callback && callback()

}

this.$element.trigger(e);

if (e.isDefaultPrevented()) return;

$tip.removeClass('in');

$.support.transition && $tip.hasClass('fade') ?

$tip

.one('bsTransitionEnd', complete)

.emulateTransitionEnd(Tooltip.TRANSITION\_DURATION) :

complete();

this.hoverState = null;

return this

};

Tooltip.prototype.fixTitle = function () {

var $e = this.$element;

if ($e.attr('title') || typeof $e.attr('data-original-title') != 'string') {

$e.attr('data-original-title', $e.attr('title') || '').attr('title', '')

}

};

Tooltip.prototype.hasContent = function () {

return this.getTitle()

};

Tooltip.prototype.getPosition = function ($element) {

$element = $element || this.$element;

var el = $element[0];

var isBody = el.tagName == 'BODY';

var elRect = el.getBoundingClientRect();

if (elRect.width == null) {

// width and height are missing in IE8, so compute them manually; see <https://github.com/twbs/bootstrap/issues/14093>

elRect = $.extend({}, elRect, {width: elRect.right - elRect.left, height: elRect.bottom - elRect.top})

}

var isSvg = window.SVGElement && el instanceof window.SVGElement;

// Avoid using $.offset() on SVGs since it gives incorrect results in jQuery 3.

// See <https://github.com/twbs/bootstrap/issues/20280>

var elOffset = isBody ? {top: 0, left: 0} : (isSvg ? null : $element.offset());

var scroll = {scroll: isBody ? document.documentElement.scrollTop || document.body.scrollTop : $element.scrollTop()};

var outerDims = isBody ? {width: $(window).width(), height: $(window).height()} : null;

return $.extend({}, elRect, scroll, outerDims, elOffset)

};

Tooltip.prototype.getCalculatedOffset = function (placement, pos, actualWidth, actualHeight) {

return placement == 'bottom' ? {top: pos.top + pos.height, left: pos.left + pos.width / 2 - actualWidth / 2} :

placement == 'top' ? {top: pos.top - actualHeight, left: pos.left + pos.width / 2 - actualWidth / 2} :

placement == 'left' ? {top: pos.top + pos.height / 2 - actualHeight / 2, left: pos.left - actualWidth} :

/\* placement == 'right' \*/ {

top: pos.top + pos.height / 2 - actualHeight / 2,

left: pos.left + pos.width

}

};

Tooltip.prototype.getViewportAdjustedDelta = function (placement, pos, actualWidth, actualHeight) {

var delta = {top: 0, left: 0};

if (!this.$viewport) return delta;

var viewportPadding = this.options.viewport && this.options.viewport.padding || 0;

var viewportDimensions = this.getPosition(this.$viewport);

if (/right|left/.test(placement)) {

var topEdgeOffset = pos.top - viewportPadding - viewportDimensions.scroll;

var bottomEdgeOffset = pos.top + viewportPadding - viewportDimensions.scroll + actualHeight;

if (topEdgeOffset < viewportDimensions.top) { // top overflow

delta.top = viewportDimensions.top - topEdgeOffset

} else if (bottomEdgeOffset > viewportDimensions.top + viewportDimensions.height) { // bottom overflow

delta.top = viewportDimensions.top + viewportDimensions.height - bottomEdgeOffset

}

} else {

var leftEdgeOffset = pos.left - viewportPadding;

var rightEdgeOffset = pos.left + viewportPadding + actualWidth;

if (leftEdgeOffset < viewportDimensions.left) { // left overflow

delta.left = viewportDimensions.left - leftEdgeOffset

} else if (rightEdgeOffset > viewportDimensions.right) { // right overflow

delta.left = viewportDimensions.left + viewportDimensions.width - rightEdgeOffset

}

}

return delta

};

Tooltip.prototype.getTitle = function () {

var title;

var $e = this.$element;

var o = this.options;

title = $e.attr('data-original-title')

|| (typeof o.title == 'function' ? o.title.call($e[0]) : o.title);

return title

};

Tooltip.prototype.getUID = function (prefix) {

do prefix += ~~(Math.random() \* 1000000);

while (document.getElementById(prefix));

return prefix

};

Tooltip.prototype.tip = function () {

if (!this.$tip) {

this.$tip = $(this.options.template);

if (this.$tip.length != 1) {

throw new Error(this.type + ' `template` option must consist of exactly 1 top-level element!')

}

}

return this.$tip

};

Tooltip.prototype.arrow = function () {

return (this.$arrow = this.$arrow || this.tip().find('.tooltip-arrow'))

};

Tooltip.prototype.enable = function () {

this.enabled = true

};

Tooltip.prototype.disable = function () {

this.enabled = false

};

Tooltip.prototype.toggleEnabled = function () {

this.enabled = !this.enabled

};

Tooltip.prototype.toggle = function (e) {

var self = this;

if (e) {

self = $(e.currentTarget).data('bs.' + this.type);

if (!self) {

self = new this.constructor(e.currentTarget, this.getDelegateOptions());

$(e.currentTarget).data('bs.' + this.type, self)

}

}

if (e) {

self.inState.click = !self.inState.click;

if (self.isInStateTrue()) self.enter(self);

else self.leave(self)

} else {

self.tip().hasClass('in') ? self.leave(self) : self.enter(self)

}

};

Tooltip.prototype.destroy = function () {

var that = this;

clearTimeout(this.timeout);

this.hide(function () {

that.$element.off('.' + that.type).removeData('bs.' + that.type);

if (that.$tip) {

that.$tip.detach()

}

that.$tip = null;

that.$arrow = null;

that.$viewport = null;

that.$element = null

})

};

// TOOLTIP PLUGIN DEFINITION

// =========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.tooltip');

var options = typeof option == 'object' && option;

if (!data && /destroy|hide/.test(option)) return;

if (!data) $this.data('bs.tooltip', (data = new Tooltip(this, options)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.tooltip;

$.fn.tooltip = Plugin;

$.fn.tooltip.Constructor = Tooltip;

// TOOLTIP NO CONFLICT

// ===================

$.fn.tooltip.noConflict = function () {

$.fn.tooltip = old;

return this

}

}(jQuery);

/\* ========================================================================

\* Bootstrap: popover.js v3.3.7

\* <http://getbootstrap.com/javascript/#popovers>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// POPOVER PUBLIC CLASS DEFINITION

// ===============================

var Popover = function (element, options) {

this.init('popover', element, options)

};

if (!$.fn.tooltip) throw new Error('Popover requires tooltip.js');

Popover.VERSION = '3.3.7';

Popover.DEFAULTS = $.extend({}, $.fn.tooltip.Constructor.DEFAULTS, {

placement: 'right',

trigger: 'click',

content: '',

template: '<div class="popover" role="tooltip"><div class="arrow"></div><h3 class="popover-title"></h3><div class="popover-content"></div></div>'

});

// NOTE: POPOVER EXTENDS tooltip.js

// ================================

Popover.prototype = $.extend({}, $.fn.tooltip.Constructor.prototype);

Popover.prototype.constructor = Popover;

Popover.prototype.getDefaults = function () {

return Popover.DEFAULTS

};

Popover.prototype.setContent = function () {

var $tip = this.tip();

var title = this.getTitle();

var content = this.getContent();

$tip.find('.popover-title')[this.options.html ? 'html' : 'text'](title);

$tip.find('.popover-content').children().detach().end()[ // we use append for html objects to maintain js events

this.options.html ? (typeof content == 'string' ? 'html' : 'append') : 'text'

](content);

$tip.removeClass('fade top bottom left right in');

// IE8 doesn't accept hiding via the `:empty` pseudo selector, we have to do

// this manually by checking the contents.

if (!$tip.find('.popover-title').html()) $tip.find('.popover-title').hide()

};

Popover.prototype.hasContent = function () {

return this.getTitle() || this.getContent()

};

Popover.prototype.getContent = function () {

var $e = this.$element;

var o = this.options;

return $e.attr('data-content')

|| (typeof o.content == 'function' ?

o.content.call($e[0]) :

o.content)

};

Popover.prototype.arrow = function () {

return (this.$arrow = this.$arrow || this.tip().find('.arrow'))

};

// POPOVER PLUGIN DEFINITION

// =========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.popover');

var options = typeof option == 'object' && option;

if (!data && /destroy|hide/.test(option)) return;

if (!data) $this.data('bs.popover', (data = new Popover(this, options)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.popover;

$.fn.popover = Plugin;

$.fn.popover.Constructor = Popover;

// POPOVER NO CONFLICT

// ===================

$.fn.popover.noConflict = function () {

$.fn.popover = old;

return this

}

}(jQuery);

/\* ========================================================================

\* Bootstrap: scrollspy.js v3.3.7

\* <http://getbootstrap.com/javascript/#scrollspy>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// SCROLLSPY CLASS DEFINITION

// ==========================

function ScrollSpy(element, options) {

this.$body = $(document.body);

this.$scrollElement = $(element).is(document.body) ? $(window) : $(element);

this.options = $.extend({}, ScrollSpy.DEFAULTS, options);

this.selector = (this.options.target || '') + ' .nav li > a';

this.offsets = [];

this.targets = [];

this.activeTarget = null;

this.scrollHeight = 0;

this.$scrollElement.on('scroll.bs.scrollspy', $.proxy(this.process, this));

this.refresh();

this.process()

}

ScrollSpy.VERSION = '3.3.7';

ScrollSpy.DEFAULTS = {

offset: 10

};

ScrollSpy.prototype.getScrollHeight = function () {

return this.$scrollElement[0].scrollHeight || Math.max(this.$body[0].scrollHeight, document.documentElement.scrollHeight)

};

ScrollSpy.prototype.refresh = function () {

var that = this;

var offsetMethod = 'offset';

var offsetBase = 0;

this.offsets = [];

this.targets = [];

this.scrollHeight = this.getScrollHeight();

if (!$.isWindow(this.$scrollElement[0])) {

offsetMethod = 'position';

offsetBase = this.$scrollElement.scrollTop()

}

this.$body

.find(this.selector)

.map(function () {

var $el = $(this);

var href = $el.data('target') || $el.attr('href');

var $href = /^#./.test(href) && $(href);

return ($href

&& $href.length

&& $href.is(':visible')

&& [[$href[offsetMethod]().top + offsetBase, href]]) || null

})

.sort(function (a, b) {

return a[0] - b[0]

})

.each(function () {

that.offsets.push(this[0]);

that.targets.push(this[1])

})

};

ScrollSpy.prototype.process = function () {

var scrollTop = this.$scrollElement.scrollTop() + this.options.offset;

var scrollHeight = this.getScrollHeight();

var maxScroll = this.options.offset + scrollHeight - this.$scrollElement.height();

var offsets = this.offsets;

var targets = this.targets;

var activeTarget = this.activeTarget;

var i;

if (this.scrollHeight != scrollHeight) {

this.refresh()

}

if (scrollTop >= maxScroll) {

return activeTarget != (i = targets[targets.length - 1]) && this.activate(i)

}

if (activeTarget && scrollTop < offsets[0]) {

this.activeTarget = null;

return this.clear()

}

for (i = offsets.length; i--;) {

activeTarget != targets[i]

&& scrollTop >= offsets[i]

&& (offsets[i + 1] === undefined || scrollTop < offsets[i + 1])

&& this.activate(targets[i])

}

};

ScrollSpy.prototype.activate = function (target) {

this.activeTarget = target;

this.clear();

var selector = this.selector +

'[data-target="' + target + '"],' +

this.selector + '[href="' + target + '"]';

var active = $(selector)

.parents('li')

.addClass('active');

if (active.parent('.dropdown-menu').length) {

active = active

.closest('li.dropdown')

.addClass('active')

}

active.trigger('activate.bs.scrollspy')

};

ScrollSpy.prototype.clear = function () {

$(this.selector)

.parentsUntil(this.options.target, '.active')

.removeClass('active')

};

// SCROLLSPY PLUGIN DEFINITION

// ===========================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.scrollspy');

var options = typeof option == 'object' && option;

if (!data) $this.data('bs.scrollspy', (data = new ScrollSpy(this, options)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.scrollspy;

$.fn.scrollspy = Plugin;

$.fn.scrollspy.Constructor = ScrollSpy;

// SCROLLSPY NO CONFLICT

// =====================

$.fn.scrollspy.noConflict = function () {

$.fn.scrollspy = old;

return this

};

// SCROLLSPY DATA-API

// ==================

$(window).on('load.bs.scrollspy.data-api', function () {

$('[data-spy="scroll"]').each(function () {

var $spy = $(this);

Plugin.call($spy, $spy.data())

})

})

}(jQuery);

/\* ========================================================================

\* Bootstrap: tab.js v3.3.7

\* <http://getbootstrap.com/javascript/#tabs>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// TAB CLASS DEFINITION

// ====================

var Tab = function (element) {

// jscs:disable requireDollarBeforejQueryAssignment

this.element = $(element)

// jscs:enable requireDollarBeforejQueryAssignment

};

Tab.VERSION = '3.3.7';

Tab.TRANSITION\_DURATION = 150;

Tab.prototype.show = function () {

var $this = this.element;

var $ul = $this.closest('ul:not(.dropdown-menu)');

var selector = $this.data('target');

if (!selector) {

selector = $this.attr('href');

selector = selector && selector.replace(/.\*(?=#[^\s]\*$)/, '') // strip for ie7

}

if ($this.parent('li').hasClass('active')) return;

var $previous = $ul.find('.active:last a');

var hideEvent = $.Event('hide.bs.tab', {

relatedTarget: $this[0]

});

var showEvent = $.Event('show.bs.tab', {

relatedTarget: $previous[0]

});

$previous.trigger(hideEvent);

$this.trigger(showEvent);

if (showEvent.isDefaultPrevented() || hideEvent.isDefaultPrevented()) return;

var $target = $(selector);

this.activate($this.closest('li'), $ul);

this.activate($target, $target.parent(), function () {

$previous.trigger({

type: 'hidden.bs.tab',

relatedTarget: $this[0]

});

$this.trigger({

type: 'shown.bs.tab',

relatedTarget: $previous[0]

})

})

};

Tab.prototype.activate = function (element, container, callback) {

var $active = container.find('> .active');

var transition = callback

&& $.support.transition

&& ($active.length && $active.hasClass('fade') || !!container.find('> .fade').length);

function next() {

$active

.removeClass('active')

.find('> .dropdown-menu > .active')

.removeClass('active')

.end()

.find('[data-toggle="tab"]')

.attr('aria-expanded', false);

element

.addClass('active')

.find('[data-toggle="tab"]')

.attr('aria-expanded', true);

if (transition) {

element[0].offsetWidth; // reflow for transition

element.addClass('in')

} else {

element.removeClass('fade')

}

if (element.parent('.dropdown-menu').length) {

element

.closest('li.dropdown')

.addClass('active')

.end()

.find('[data-toggle="tab"]')

.attr('aria-expanded', true)

}

callback && callback()

}

$active.length && transition ?

$active

.one('bsTransitionEnd', next)

.emulateTransitionEnd(Tab.TRANSITION\_DURATION) :

next();

$active.removeClass('in')

};

// TAB PLUGIN DEFINITION

// =====================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.tab');

if (!data) $this.data('bs.tab', (data = new Tab(this)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.tab;

$.fn.tab = Plugin;

$.fn.tab.Constructor = Tab;

// TAB NO CONFLICT

// ===============

$.fn.tab.noConflict = function () {

$.fn.tab = old;

return this

};

// TAB DATA-API

// ============

var clickHandler = function (e) {

e.preventDefault();

Plugin.call($(this), 'show')

};

$(document)

.on('click.bs.tab.data-api', '[data-toggle="tab"]', clickHandler)

.on('click.bs.tab.data-api', '[data-toggle="pill"]', clickHandler)

}(jQuery);

/\* ========================================================================

\* Bootstrap: affix.js v3.3.7

\* <http://getbootstrap.com/javascript/#affix>

\* ========================================================================

\* Copyright 2011-2016 Twitter, Inc.

\* Licensed under MIT (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

\* ======================================================================== \*/

+function ($) {

'use strict';

// AFFIX CLASS DEFINITION

// ======================

var Affix = function (element, options) {

this.options = $.extend({}, Affix.DEFAULTS, options);

this.$target = $(this.options.target)

.on('scroll.bs.affix.data-api', $.proxy(this.checkPosition, this))

.on('click.bs.affix.data-api', $.proxy(this.checkPositionWithEventLoop, this));

this.$element = $(element);

this.affixed = null;

this.unpin = null;

this.pinnedOffset = null;

this.checkPosition()

};

Affix.VERSION = '3.3.7';

Affix.RESET = 'affix affix-top affix-bottom';

Affix.DEFAULTS = {

offset: 0,

target: window

};

Affix.prototype.getState = function (scrollHeight, height, offsetTop, offsetBottom) {

var scrollTop = this.$target.scrollTop();

var position = this.$element.offset();

var targetHeight = this.$target.height();

if (offsetTop != null && this.affixed == 'top') return scrollTop < offsetTop ? 'top' : false;

if (this.affixed == 'bottom') {

if (offsetTop != null) return (scrollTop + this.unpin <= position.top) ? false : 'bottom';

return (scrollTop + targetHeight <= scrollHeight - offsetBottom) ? false : 'bottom'

}

var initializing = this.affixed == null;

var colliderTop = initializing ? scrollTop : position.top;

var colliderHeight = initializing ? targetHeight : height;

if (offsetTop != null && scrollTop <= offsetTop) return 'top';

if (offsetBottom != null && (colliderTop + colliderHeight >= scrollHeight - offsetBottom)) return 'bottom';

return false

};

Affix.prototype.getPinnedOffset = function () {

if (this.pinnedOffset) return this.pinnedOffset;

this.$element.removeClass(Affix.RESET).addClass('affix');

var scrollTop = this.$target.scrollTop();

var position = this.$element.offset();

return (this.pinnedOffset = position.top - scrollTop)

};

Affix.prototype.checkPositionWithEventLoop = function () {

setTimeout($.proxy(this.checkPosition, this), 1)

};

Affix.prototype.checkPosition = function () {

if (!this.$element.is(':visible')) return;

var height = this.$element.height();

var offset = this.options.offset;

var offsetTop = offset.top;

var offsetBottom = offset.bottom;

var scrollHeight = Math.max($(document).height(), $(document.body).height());

if (typeof offset != 'object') offsetBottom = offsetTop = offset;

if (typeof offsetTop == 'function') offsetTop = offset.top(this.$element);

if (typeof offsetBottom == 'function') offsetBottom = offset.bottom(this.$element);

var affix = this.getState(scrollHeight, height, offsetTop, offsetBottom);

if (this.affixed != affix) {

if (this.unpin != null) this.$element.css('top', '');

var affixType = 'affix' + (affix ? '-' + affix : '');

var e = $.Event(affixType + '.bs.affix');

this.$element.trigger(e);

if (e.isDefaultPrevented()) return;

this.affixed = affix;

this.unpin = affix == 'bottom' ? this.getPinnedOffset() : null;

this.$element

.removeClass(Affix.RESET)

.addClass(affixType)

.trigger(affixType.replace('affix', 'affixed') + '.bs.affix')

}

if (affix == 'bottom') {

this.$element.offset({

top: scrollHeight - height - offsetBottom

})

}

};

// AFFIX PLUGIN DEFINITION

// =======================

function Plugin(option) {

return this.each(function () {

var $this = $(this);

var data = $this.data('bs.affix');

var options = typeof option == 'object' && option;

if (!data) $this.data('bs.affix', (data = new Affix(this, options)));

if (typeof option == 'string') data[option]()

})

}

var old = $.fn.affix;

$.fn.affix = Plugin;

$.fn.affix.Constructor = Affix;

// AFFIX NO CONFLICT

// =================

$.fn.affix.noConflict = function () {

$.fn.affix = old;

return this

};

// AFFIX DATA-API

// ==============

$(window).on('load', function () {

$('[data-spy="affix"]').each(function () {

var $spy = $(this);

var data = $spy.data();

data.offset = data.offset || {};

if (data.offsetBottom != null) data.offset.bottom = data.offsetBottom;

if (data.offsetTop != null) data.offset.top = data.offsetTop;

Plugin.call($spy, data)

})

})

}(jQuery);

(function ($) {

$.toggleShowPassword = function (options) {

var settings = $.extend({

field: "#password",

control: "#toggle\_show\_password",

}, options);

var control = $(settings.control);

var field = $(settings.field);

control.bind('click', function () {

if (control.is(':checked')) {

field.attr('type', 'text');

} else {

field.attr('type', 'password');

}

})

};

$.transferDisplay = function () {

$("#transferFrom").change(function () {

if ($("#transferFrom").val() == 'Primary') {

$('#transferTo').val('Savings');

} else if ($("#transferFrom").val() == 'Savings') {

$('#transferTo').val('Primary');

}

});

$("#transferTo").change(function () {

if ($("#transferTo").val() == 'Primary') {

$('#transferFrom').val('Savings');

} else if ($("#transferTo").val() == 'Savings') {

$('#transferFrom').val('Primary');

}

});

};

}(jQuery));

$(document).ready(function () {

var confirm = function () {

bootbox.confirm({

title: "Appointment Confirmation",

message: "Do you really want to schedule this appointment?",

buttons: {

cancel: {

label: '<i class="fa fa-times"></i> Cancel'

},

confirm: {

label: '<i class="fa fa-check"></i> Confirm'

}

},

callback: function (result) {

if (result == true) {

$('#appointmentForm').submit();

} else {

console.log("Scheduling cancelled.");

}

}

});

};

$.toggleShowPassword({

field: '#password',

control: "#showPassword"

});

$.transferDisplay();

$(".form\_datetime").datetimepicker({

format: "yyyy-mm-dd hh:mm",

autoclose: true,

todayBtn: true,

startDate: "2013-02-14 10:00",

minuteStep: 10

});

$('#submitAppointment').click(function () {

confirm();

});

});