Post Request with image -

@interface lodgeComplaint ()<UINavigationControllerDelegate, UIImagePickerControllerDelegate>{

UIImage \*selectedImage1;

NSData \*imageData1;

}

// MARK: Add Image

- (IBAction)addImageClicked:(UIButton \*)sender {

[[UIApplication sharedApplication] sendAction:@selector(resignFirstResponder) to:nil from:nil forEvent:nil];

UIAlertController \*alert = [UIAlertController alertControllerWithTitle:@"Select Type" message:@"" preferredStyle:UIAlertControllerStyleAlert];

UIAlertAction \*camera = [UIAlertAction actionWithTitle:@"Camera" style:UIAlertActionStyleDefault handler:^(UIAlertAction \* \_Nonnull action) {

UIImagePickerController \*pickerView = [[UIImagePickerController alloc] init];

pickerView.allowsEditing = YES;

pickerView.delegate = self;

[pickerView setSourceType:UIImagePickerControllerSourceTypeCamera];

[self presentViewController:pickerView animated:YES completion:nil];

}];

[alert addAction:camera];

UIAlertAction \*gallery = [UIAlertAction actionWithTitle:@"Gallery" style:UIAlertActionStyleDefault handler:^(UIAlertAction \* \_Nonnull action) {

UIImagePickerController \*pickerView = [[UIImagePickerController alloc] init];

pickerView.allowsEditing = YES;

pickerView.delegate = self;

[pickerView setSourceType:UIImagePickerControllerSourceTypePhotoLibrary];

[self presentViewController:pickerView animated:YES completion:nil];

}];

[alert addAction:gallery];

UIAlertAction \*cancel = [UIAlertAction actionWithTitle:@"Cancel" style:UIAlertActionStyleDestructive handler:^(UIAlertAction \* \_Nonnull action) {

}];

[alert addAction:cancel];

[self presentViewController:alert animated:YES completion:nil];

}

- (void)imagePickerController:(UIImagePickerController \*)picker didFinishPickingMediaWithInfo:(NSDictionary \*)info{

selectedImage1 = [info objectForKey:@"UIImagePickerControllerOriginalImage"];

\_iAddImage.image = selectedImage1;

imageData1 = UIImageJPEGRepresentation(selectedImage1, 2.0);

[picker dismissViewControllerAnimated:YES completion:NULL];

\_iAttach.hidden = NO;

}

NSUserDefaults \*defaults = [NSUserDefaults standardUserDefaults];

NSDictionary \*paramDict = [NSDictionary dictionaryWithObjectsAndKeys:[,nil];

NSURLSessionConfiguration \*configuration = [NSURLSessionConfiguration defaultSessionConfiguration];

NSURLSession \*session = [NSURLSession sessionWithConfiguration:configuration delegate:nil delegateQueue:nil];

NSMutableURLRequest \*request = [[NSMutableURLRequest alloc] init];

[request setHTTPShouldHandleCookies:NO];

[request setTimeoutInterval:60];

[request setHTTPMethod:@"POST"];

NSString \*boundary = @"------";

NSString \*contentType = [NSString stringWithFormat:@"multipart/form-data; boundary=%@", boundary];

[request setValue:contentType forHTTPHeaderField: @"Content-Type"];

NSMutableData \*body = [NSMutableData data];

for (NSString \*param in paramDict) {

[body appendData:[[NSString stringWithFormat:@"--%@\r\n", boundary] dataUsingEncoding:NSUTF8StringEncoding]];

[body appendData:[[NSString stringWithFormat:@"Content-Disposition: form-data; name=\"%@\"\r\n\r\n", param] dataUsingEncoding:NSUTF8StringEncoding]];

[body appendData:[[NSString stringWithFormat:@"%@\r\n", [paramDict objectForKey:param]] dataUsingEncoding:NSUTF8StringEncoding]];

}

NSString \*FileParamConstant1 = @"";

if (imageData1)

{

[body appendData:[[NSString stringWithFormat:@"--%@\r\n", boundary] dataUsingEncoding:NSUTF8StringEncoding]];

[body appendData:[[NSString stringWithFormat:@"Content-Disposition: form-data; name=\"%@\"; filename=\"image.jpg\"\r\n", FileParamConstant1] dataUsingEncoding:NSUTF8StringEncoding]];

[body appendData:[@"Content-Type:image/jpeg\r\n\r\n" dataUsingEncoding:NSUTF8StringEncoding]];

[body appendData:imageData1];

[body appendData:[[NSString stringWithFormat:@"\r\n"] dataUsingEncoding:NSUTF8StringEncoding]];

}

[body appendData:[[NSString stringWithFormat:@"--%@--\r\n", boundary] dataUsingEncoding:NSUTF8StringEncoding]];

[request setHTTPBody:body];

NSURL \*url = [NSURL URLWithString:@""];

[request setURL:url];

NSURLSessionDataTask \*postDataTask = [session dataTaskWithRequest:request completionHandler:^(NSData \*data, NSURLResponse \*response, NSError \*error){

dispatch\_async(dispatch\_get\_main\_queue(), ^{

if(data == nil || [data isKindOfClass:[NSNull class]]){

[hud hideAnimated:YES];

UIAlertController \*alert = [UIAlertController alertControllerWithTitle:@"Error!!" message:@"There was an error connecting the server. Please wait for some time and try again later." preferredStyle:UIAlertControllerStyleAlert];

UIAlertAction \*ok = [UIAlertAction actionWithTitle:@"OK" style:UIAlertActionStyleDestructive handler:^(UIAlertAction \* \_Nonnull action) {

}];

[alert addAction:ok];

[self presentViewController:alert animated:YES completion:nil];

}

else{

[hud hideAnimated:YES];

NSDictionary \*responseData = [NSJSONSerialization JSONObjectWithData:data options:0 error:NULL];

if([[responseData valueForKey:@"Result"] isEqualToString:@"true"]){

UIAlertController \*alert = [UIAlertController alertControllerWithTitle:@"Done" message:@"Successfully." preferredStyle:UIAlertControllerStyleAlert];

UIAlertAction \*ok = [UIAlertAction actionWithTitle:@"OK" style:UIAlertActionStyleDefault handler:^(UIAlertAction \*action) {

[self.navigationController popViewControllerAnimated:YES];

}];

[alert addAction:ok];

[self presentViewController:alert animated:YES completion:nil];

}

else{

UIAlertController \*alert = [UIAlertController alertControllerWithTitle:@"Error !!" message:@"An error occured. Please try again later" preferredStyle:UIAlertControllerStyleAlert];

UIAlertAction \*ok = [UIAlertAction actionWithTitle:@"OK" style:UIAlertActionStyleDestructive handler:^(UIAlertAction \*action) {

}];

[alert addAction:ok];

[self presentViewController:alert animated:YES completion:nil];

}

}

});

}];

[postDataTask resume];