

/home/français/android/Comment puis-je envoyer des emails à partir de mon application Android?_

Comment puis-je envoyer des emails à partir de mon application Android?

Je suis en train d'écrire une application pour Android. Comment puis-je envoyer un email à partir de celui-ci?

 **android email**

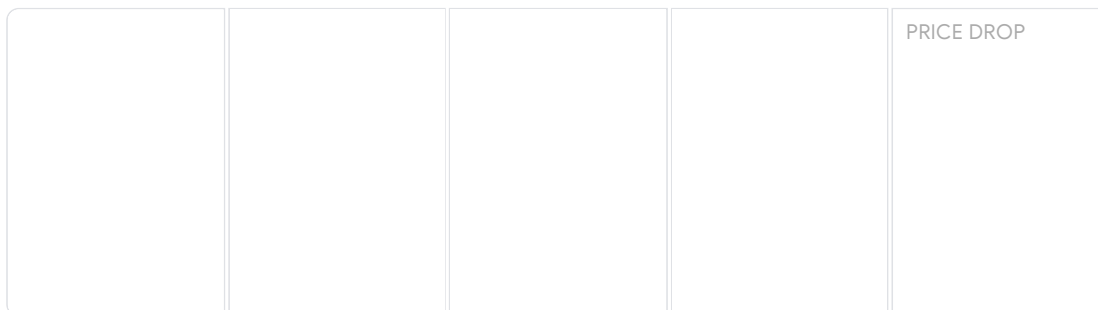
 486



Rakesh



4 févr. 2010



Outv
Lady
Berrylc

La meilleure (et la plus simple) consiste à utiliser une `Intent`:



```
Intent i = new Intent(Intent.ACTION_SEND);
i.setType("message/rfc822");
i.putExtra(Intent.EXTRA_EMAIL, new String[]{"recipient@example.com"});
i.putExtra(Intent.EXTRA_SUBJECT, "subject of email");
i.putExtra(Intent.EXTRA_TEXT, "body of email");
try {
    startActivity(Intent.createChooser(i, "Send mail..."));
} catch (Android.content.ActivityNotFoundException ex) {
    Toast.makeText(MyActivity.this, "There are no email clients installed.", Toast.LENGTH_SHORT).show();
}
```

Sinon, vous devrez écrire votre propre [it-swarm-fr.com](https://www.it-swarm-fr.com)

 928



Jeremy Logan



4 févr. 2010



Utilisez `.setType("message/rfc822")` ou le sélecteur vous montrera toutes les applications (nombreuses) prenant en charge l'intention d'envoi.



Je l'utilise depuis longtemps et cela semble bien, aucune application non-mail ne s'affiche Juste un autre moyen d'envoyer un e-mail d'intention:

```
Intent intent = new Intent(Intent.ACTION_SENDTO); // it's not ACTION_SEND
intent.setType("text/plain");
intent.putExtra(Intent.EXTRA_SUBJECT, "Subject of email");
intent.putExtra(Intent.EXTRA_TEXT, "Body of email");
intent.setData(Uri.parse("mailto:default@recipient.com")); // or just "mailto:" for blank
intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK); // this will make such that when user returns to your app, your app is
displayed, instead of the email app.
startActivity(intent);
```



J'utilisais quelque chose comme la réponse actuellement acceptée pour envoyer des courriels avec un fichier journal d'erreur binaire joint. GMail et K-9 l'envoient très bien et tout se passe bien sur mon serveur de messagerie. Le seul problème était mon client de messagerie de choix, Thunderbird, qui avait des problèmes pour ouvrir/enregistrer le fichier journal joint. En fait, il n'a tout simplement pas sauvegardé le fichier sans se plaindre.

J'ai jeté un coup d'œil à l'un des codes sources de ce courrier et j'ai remarqué que la pièce jointe au fichier journal contenait (naturellement) le type mime `message/rfc822`. Bien sûr, cette pièce jointe n'est pas un courriel attaché. Mais Thunderbird ne peut pas gérer cette petite erreur avec grâce. C'était donc un peu nul.

Après quelques recherches et expérimentations, j'ai trouvé la solution suivante:

```
public Intent createEmailOnlyChooserIntent(Intent source,
    CharSequence chooserTitle) {
```

```

Stack<Intent> intents = new Stack<Intent>();
Intent i = new Intent(Intent.ACTION_SENDTO, Uri.fromParts("mailto",
    "info@domain.com", null));
List<ResolveInfo> activities = getPackageManager()
    .queryIntentActivities(i, 0);

for(ResolveInfo ri : activities) {
    Intent target = new Intent(source);
    target.setPackage(ri.activityInfo.packageName);
    intents.add(target);
}

if(!intents.isEmpty()) {
    Intent chooserIntent = Intent.createChooser(intents.remove(0),
        chooserTitle);
    chooserIntent.putExtra(Intent.EXTRA_INITIAL_INTENTS,
        intents.toArray(new Parcelable[intents.size()]));

    return chooserIntent;
} else {
    return Intent.createChooser(source, chooserTitle);
}
}

```

Il peut être utilisé comme suit:

```

Intent i = new Intent(Intent.ACTION_SEND);
i.setType("*/*");
i.putExtra(Intent.EXTRA_STREAM, Uri.fromFile(crashLogFile));
i.putExtra(Intent.EXTRA_EMAIL, new String[] {
    Android_SUPPORT_EMAIL
});
i.putExtra(Intent.EXTRA_SUBJECT, "Crash report");
i.putExtra(Intent.EXTRA_TEXT, "Some crash report details");

startActivity(createEmailOnlyChooserIntent(i, "Send via email"));

```

Comme vous pouvez le constater, la méthode `createEmailOnlyChooserIntent` peut être facilement alimentée avec l'intention et le type mime corrects.

Il passe ensuite en revue la liste des activités disponibles qui répondent à une intention de protocole `ACTION_SENDTO` `mailto` (qui sont uniquement des applications de messagerie électronique) et construit un sélecteur basé sur cette liste d'activités et l'intention `ACTION_SEND` d'origine avec le type mime correct.

Un autre avantage est que Skype ne figure plus dans la liste (ce qui correspond au type mime `rfc822`).



49



tiguchi



9 oct. 2012

Pour **JUST LET EMAIL APPS** pour résoudre votre problème, vous devez spécifier `ACTION_SENDTO` en tant qu'action et `mailto` en tant que données.

```

private void sendEmail(){

    Intent emailIntent = new Intent(Intent.ACTION_SENDTO);

```

```
emailIntent.setData(Uri.parse("mailto:" + "recipient@example.com"));
emailIntent.putExtra(Intent.EXTRA_SUBJECT, "My email's subject");
emailIntent.putExtra(Intent.EXTRA_TEXT, "My email's body");

try {
    startActivity(Intent.createChooser(emailIntent, "Send email using..."));
} catch (Android.content.ActivityNotFoundException ex) {
    Toast.makeText(Activity.this, "No email clients installed.", Toast.LENGTH_SHORT).show();
}

}
```



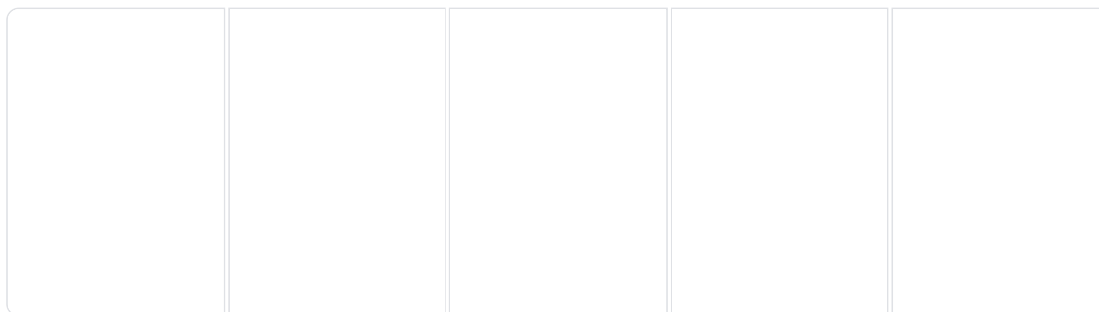
33



wildzic



2 avr. 2014



Outv
Lady
Berrylc

La stratégie d'utilisation de `.setType("message/rfc822")` ou `ACTION_SEND` semble également correspondre aux applications qui ne sont pas des clients de messagerie, telles que *Android Beam* et *Bluetooth*.

Utiliser `ACTION_SENDTO` et un `mailto:` URI semble fonctionner parfaitement, et [est recommandé dans la documentation destinée aux développeurs](#). Cependant, si vous faites cela sur les émulateurs officiels et qu'il n'y a pas de comptes de messagerie configurés (ou s'il n'y a pas de clients de messagerie), vous obtenez l'erreur suivante:



Action non supportée

Cette action n'est actuellement pas prise en charge.

Comme indiqué ci-dessous:

Il se trouve que les émulateurs résolvent l'intention en une activité appelée `com.Android.fallback.Fallback`, qui affiche le message ci-dessus. [Apparemment c'est par conception.](#)

Si vous voulez que votre application contourne cela afin qu'elle fonctionne également correctement sur les émulateurs officiels, vous pouvez la vérifier avant d'essayer d'envoyer l'e-mail:

```
private void sendEmail() {
    Intent intent = new Intent(Intent.ACTION_SENDTO)
        .setData(new Uri.Builder().scheme("mailto").build())
        .putExtra(Intent.EXTRA_EMAIL, new String[] { "John Smith <johnsmith@yourdomain.com>" })
        .putExtra(Intent.EXTRA_SUBJECT, "Email subject")
        .putExtra(Intent.EXTRA_TEXT, "Email body")
    ;
}
```

```

ComponentName emailApp = intent.resolveActivity(getPackageManager());
ComponentName unsupportedAction = ComponentName.unflattenFromString("com.Android.fallback/.Fallback");
if (emailApp != null && !emailApp.equals(unsupportedAction))
    try {
        // Needed to customise the chooser dialog title since it might default to "Share with"
        // Note that the chooser will still be skipped if only one app is matched
        Intent chooser = Intent.createChooser(intent, "Send email with");
        startActivity(chooser);
        return;
    }
    catch (ActivityNotFoundException ignored) {
    }

    Toast
        .makeText(this, "Couldn't find an email app and account", Toast.LENGTH_LONG)
        .show();
}

```

Trouvez plus d'informations dans [la documentation pour les développeurs](#) .



18



Sam



25 juin 2015

J'ai résolu ce problème avec de simples lignes de code, comme l'explique la documentation Android.

(<https://developer.Android.com/guide/components/intents-common.html#Email>)

Le plus important est le drapeau: il s'agit de `ACTION_SENDTO` et non de `ACTION_SEND`

L'autre ligne importante est

```
intent.setData(Uri.parse("mailto:")); ****// only email apps should handle this****
```

À propos, si vous envoyez une `Extra` vide, la `if()` à la fin ne fonctionnera pas et l'application ne lancera pas le client de messagerie.

Selon la documentation Android. Si vous voulez vous assurer que votre intention est gérée uniquement par une application de messagerie (et non par d'autres applications de messagerie texte ou sociales), utilisez l'action `ACTION_SENDTO` et incluez le schéma de données `"mailto:"`. Par exemple:

```

public void composeEmail(String[] addresses, String subject) {
    Intent intent = new Intent(Intent.ACTION_SENDTO);
    intent.setData(Uri.parse("mailto:")); // only email apps should handle this
    intent.putExtra(Intent.EXTRA_EMAIL, addresses);
    intent.putExtra(Intent.EXTRA_SUBJECT, subject);
    if (intent.resolveActivity(getPackageManager()) != null) {
        startActivity(intent);
    }
}

```



18



Pedro Varela



21 juil. 2015

L'envoi de courrier électronique peut se faire avec des intentions qui ne nécessiteront aucune configuration. Mais cela nécessitera une interaction de l'utilisateur et la mise en page sera un peu restreinte.

Construire et envoyer un courrier électronique plus complexe sans interaction de l'utilisateur implique la création de votre propre client. La première chose est que les API Java Sun Java pour la messagerie électronique ne sont pas disponibles. J'ai réussi à utiliser la bibliothèque Apache Mime4j pour créer des e-mails. Tous basés sur les docs à [nilvec](#).



12



Rene



4 févr. 2010

simple essayez celui-ci

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    buttonSend = (Button) findViewById(R.id.buttonSend);
    textTo = (EditText) findViewById(R.id.editTextTo);
    textSubject = (EditText) findViewById(R.id.editTextSubject);
    textMessage = (EditText) findViewById(R.id.editTextMessage);

    buttonSend.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View v) {

            String to = textTo.getText().toString();
            String subject = textSubject.getText().toString();
            String message = textMessage.getText().toString();

            Intent email = new Intent(Intent.ACTION_SEND);
            email.putExtra(Intent.EXTRA_EMAIL, new String[] { to });
            // email.putExtra(Intent.EXTRA_CC, new String[] { to });
            // email.putExtra(Intent.EXTRA_BCC, new String[] { to });
            email.putExtra(Intent.EXTRA_SUBJECT, subject);
            email.putExtra(Intent.EXTRA_TEXT, message);

            // need this to prompts email client only
            email.setType("message/rfc822");

            startActivity(Intent.createChooser(email, "Choose an Email client :"));

        }
    });
}
```



3



NagarjunaReddy



4 oct. 2012

Voici l'exemple de code de travail qui ouvre **application de messagerie** sur un appareil Android et rempli automatiquement avec **Adresse** et **Sujet** dans le courrier de composition.

```
protected void sendEmail() {  
    Intent intent = new Intent(Intent.ACTION_SENDTO);  
    intent.setData(Uri.parse("mailto:feedback@gmail.com"));  
    intent.putExtra(Intent.EXTRA_SUBJECT, "Feedback");  
    if (intent.resolveActivity(getPackageManager()) != null) {  
        startActivity(intent);  
    }  
}
```



3



Sridhar Nalam



16 janv. 2018

J'utilise le code ci-dessous dans mes applications. Cela montre exactement les applications client de messagerie, telles que Gmail.

```
Intent contactIntent = new Intent(Intent.ACTION_SENDTO, Uri.fromParts("mailto", getString(R.string.email_to), null));  
contactIntent.putExtra(Intent.EXTRA_SUBJECT, getString(R.string.email_subject));  
startActivity(Intent.createChooser(contactIntent, getString(R.string.email_chooser)));
```



3



lomza



20 sept. 2016

Une autre solution peut être

```
Intent emailIntent = new Intent(Android.content.Intent.ACTION_SEND);  
emailIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
emailIntent.setType("plain/text");  
emailIntent.setClassName("com.google.android.gm", "com.google.android.gm.ComposeActivityGmail");  
emailIntent.putExtra(Android.content.Intent.EXTRA_EMAIL, new String[]{"someone@gmail.com"});  
emailIntent.putExtra(Android.content.Intent.EXTRA_SUBJECT, "Yo");  
emailIntent.putExtra(Android.content.Intent.EXTRA_TEXT, "Hi");  
startActivity(emailIntent);
```

En supposant que l'application GMail soit déjà installée sur la plupart des appareils Android.



2



silentsudo



28 déc. 2015

Cette fonction est la première fonction d'intention directe de Gmail pour l'envoi de courrier électronique. Si vous ne trouvez pas gmail, activez le sélecteur d'intention. J'ai utilisé cette fonction dans de nombreuses applications commerciales et cela fonctionne bien. J'espère que cela vous aidera:

```
public static void sendEmail(Context mContext, String[] addresses, String subject, String body) {

    try {
        Intent sendIntentGmail = new Intent(Intent.ACTION_VIEW);
        sendIntentGmail.setType("plain/text");
        sendIntentGmail.setData(Uri.parse(TextUtils.join(",", addresses)));
        sendIntentGmail.setClassName("com.google.Android.gm", "com.google.Android.gm.ComposeActivityGmail");
        sendIntentGmail.putExtra(Intent.EXTRA_EMAIL, addresses);
        if (subject != null) sendIntentGmail.putExtra(Intent.EXTRA_SUBJECT, subject);
        if (body != null) sendIntentGmail.putExtra(Intent.EXTRA_TEXT, body);
        mContext.startActivity(sendIntentGmail);
    } catch (Exception e) {
        //When Gmail App is not installed or disable
        Intent sendIntentIfGmailFail = new Intent(Intent.ACTION_SEND);
        sendIntentIfGmailFail.setType("*/*");
        sendIntentIfGmailFail.putExtra(Intent.EXTRA_EMAIL, addresses);
        if (subject != null) sendIntentIfGmailFail.putExtra(Intent.EXTRA_SUBJECT, subject);
        if (body != null) sendIntentIfGmailFail.putExtra(Intent.EXTRA_TEXT, body);
        if (sendIntentIfGmailFail.resolveActivity(mContext.getPackageManager()) != null) {
            mContext.startActivity(sendIntentIfGmailFail);
        }
    }
}
```



2



Shohan Ahmed Sijan



16 mai 2017

Utilisez ceci pour envoyer un email ...

```
boolean success = EmailIntentBuilder.from(activity)
    .to("support@example.org")
    .cc("developer@example.org")
    .subject("Error report")
    .body(buildErrorReport())
    .start();
```

utiliser le gradé de construction:

```
compile 'de.cketti.mailto:email-intent-builder:1.0.0'
```



2



Manish



24 oct. 2016

Cela vous montrera que les clients de messagerie (ainsi que Paypal pour une raison inconnue)


```
public void composeEmail() {  
  
    Intent intent = new Intent(Intent.ACTION_SENDTO);  
    intent.setData(Uri.parse("mailto:"));  
    intent.putExtra(Intent.EXTRA_EMAIL, new String[]{"hi@example.com"});  
    intent.putExtra(Intent.EXTRA_SUBJECT, "Subject");  
    intent.putExtra(Intent.EXTRA_TEXT, "Body");  
    try {  
        startActivity(Intent.createChooser(intent, "Send mail..."));  
    } catch (Android.content.ActivityNotFoundException ex) {  
        Toast.makeText(MainActivity.this, "There are no email clients installed.", Toast.LENGTH_SHORT).show();  
    }  
}
```



2



Avi Parshan



30 janv. 2017

Voici comment je l'ai fait. Sympa et simple.

```
String emailUrl = "mailto:email@example.com?subject=Subject Text&body=Body Text";  
Intent request = new Intent(Intent.ACTION_VIEW);  
request.setData(Uri.parse(emailUrl));  
startActivity(request);
```



1



Dog Lover



15 juin 2016

```
Intent emailIntent = new Intent(Intent.ACTION_SENDTO, Uri.fromParts(  
    "mailto", "ebgsoldier@gmail.com", null));  
emailIntent.putExtra(Intent.EXTRA_SUBJECT, "Forgot Password");  
emailIntent.putExtra(Intent.EXTRA_TEXT, "Write your Pubg user name or Phone Number");  
startActivity(Intent.createChooser(emailIntent, "Send email..."));**strong text**
```



0



Tousif Akram



14 juil. 2019

J'ai utilisé ce code pour envoyer des messages en lançant directement la section de composition de l'application de messagerie par défaut.

```
Intent i = new Intent(Intent.ACTION_SENDTO);
i.setType("message/rfc822");
i.setData(Uri.parse("mailto:"));
i.putExtra(Intent.EXTRA_EMAIL, new String[]{"test@gmail.com"});
i.putExtra(Intent.EXTRA_SUBJECT, "Subject");
i.putExtra(Intent.EXTRA_TEXT, "body of email");
try {
    startActivity(Intent.createChooser(i, "Send mail..."));
} catch (Android.content.ActivityNotFoundException ex) {
    Toast.makeText(this, "There are no email clients installed.", Toast.LENGTH_SHORT).show();
}
```



0



Faris Muhammed



29 avr. 2018

Cette méthode fonctionne pour moi. Il ouvre l'application Gmail (si installée) et définit mailto.

```
public void openGmail(Activity activity) {
    Intent emailIntent = new Intent(Intent.ACTION_VIEW);
    emailIntent.setType("text/plain");
    emailIntent.setType("message/rfc822");
    emailIntent.setData(Uri.parse("mailto:"+activity.getString(R.string.mail_to)));
    emailIntent.putExtra(Intent.EXTRA_SUBJECT, activity.getString(R.string.app_name) + " - info ");
    final PackageManager pm = activity.getPackageManager();
    final List<ResolveInfo> matches = pm.queryIntentActivities(emailIntent, 0);
    ResolveInfo best = null;
    for (final ResolveInfo info : matches)
        if (info.activityInfo.packageName.endsWith(".gm") || info.activityInfo.name.toLowerCase().contains("gmail"))
            best = info;
    if (best != null)
        emailIntent.setClassName(best.activityInfo.packageName, best.activityInfo.name);
    activity.startActivity(emailIntent);
}
```



0



psycho



26 nov. 2017

Essaye ça:

```
String mailto = "mailto:bob@example.org" +
    "?cc=" + "alice@example.com" +
    "&subject=" + Uri.encode(subject) +
    "&body=" + Uri.encode(bodyText);

Intent emailIntent = new Intent(Intent.ACTION_SENDTO);
emailIntent.setData(Uri.parse(mailto));
```

```
try {
    startActivity(emailIntent);
} catch (ActivityNotFoundException e) {
    //TODO: Handle case where no email app is available
}
```

Le code ci-dessus ouvrira le client de messagerie préféré des utilisateurs, prérempli avec le courrier électronique prêt à être envoyé.

Source



0



Dan Bray



8 août 2019

```
/**
 * Will start the chosen Email app
 *
 * @param context    current component context.
 * @param emails     Emails you would like to send to.
 * @param subject    The subject that will be used in the Email app.
 * @param forceGmail True - if you want to open Gmail app, False otherwise. If the Gmail
 *                  app is not installed on this device a chooser will be shown.
 */
public static void sendEmail(Context context, String[] emails, String subject, boolean forceGmail) {

    Intent i = new Intent(Intent.ACTION_SENDTO);
    i.setData(Uri.parse("mailto:"));
    i.putExtra(Intent.EXTRA_EMAIL, emails);
    i.putExtra(Intent.EXTRA_SUBJECT, subject);
    if (forceGmail && isPackageInstalled(context, "com.google.Android.gm")) {
        i.setPackage("com.google.Android.gm");
        i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        context.startActivity(i);
    } else {
        try {
            context.startActivity(Intent.createChooser(i, "Send mail..."));
        } catch (ActivityNotFoundException e) {
            Toast.makeText(context, "No email app is installed on your device...", Toast.LENGTH_SHORT).show();
        }
    }
}

/**
 * Check if the given app is installed on this devuice.
 *
 * @param context    current component context.
```

```
* @param packageName The package name you would like to check.
* @return True if this package exist, otherwise False.
*/
public static boolean isPackageInstalled(@NonNull Context context, @NonNull String packageName) {
    PackageManager pm = context.getPackageManager();
    if (pm != null) {
        try {
            pm.getPackageInfo(packageName, 0);
            return true;
        } catch (PackageManager.NameNotFoundException e) {
            e.printStackTrace();
        }
    }
    return false;
}
```



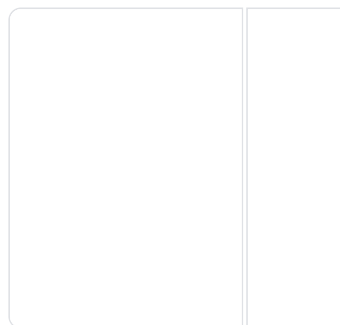
0



Gal Rom



21 déc. 2018



Outwear for Fashion La
Berrylook.com

[Envoyer un courriel](#)

[Envoi d'un courrier électronique avec pièces jointes par programme sur Android](#)

[Les tableaux 100% largeur ne fonctionnent pas dans Gmail Android](#)

[Essayer de joindre un fichier de la carte SD à un courrier électronique](#)

[Comment envoyer un email HTML](#)

[Comment ouvrir une application installée à partir d'une URL de messagerie dans Android?](#)

[Comment envoyer directement un courrier électronique depuis une application Android sans afficher le formulaire de courrier électronique par défaut du périphérique](#)

[Utilisation d'Android Intent.ACTION_SEND pour l'envoi de courrier électronique](#)

[Partage d'URL vers Facebook, Twitter et email dans Android?](#)

[Fermer/cacher le clavier virtuel Android](#)

[Android Studio: "Veuillez sélectionner Android SDK"](#)

[Erreur lors du lancement d'AVD avec un processeur AMD](#)

[Erreur lors du lancement d'ADB avec un processeur ARM](#)

[Cas d'utilisation appropriés pour Android UserManager.isUserAGoat \(\)?](#)

[Vous devez utiliser un thème \(ou un descendant\) Theme.AppCompat avec cette activité.](#)

[Obtenir l'heure et la date sur Android](#)

[L'API 'variant.getExternalNativeBuildTasks \(\)' est obsolète et a été remplacé par variante.getExternalNativeBuildProviders \(\)](#)

[AsyncTask exemple Android](#)

[Accepter automatiquement toutes les licences SDK](#)

[RecyclerView onClick](#)

[Conflit avec la dépendance 'com.Android.support:support-annotations' dans le projet ': app'. Les versions résolues pour app \(26.1.0\) et test \(27.1.1\) diffèrent.](#)



android — Comment ouvrir le Google Play Store directement à partir de mon application Android?

outlook-web-access — Comment désactiver le mode "conversation" dans Outlook Web Access?

python — Lecture d'e-mails à partir d'Outlook avec Python via MAPI

linux — Envoi d'un mail à partir d'un script shell Linux

python — Comment envoyer un email avec Python?

microsoft-outlook — Pourquoi l'icône d'enveloppe de la barre des tâches Outlook reste parfois...

microsoft-outlook — Message d'erreur Outlook lorsque vous essayez de déplacer des courriels...

gmail — Boîte de réception Gmail, tous les messages archivés par erreur

java — L'envoi de courrier SMTP ne fonctionne pas pour office365

android — Avertissements Votre Apk utilise des autorisations qui nécessitent une politique d'

android — 'adb' n'est pas reconnu comme une commande interne, externe, un programme ou un fichier de commande

email — Pourquoi est-il difficile d'ouvrir un e-mail suspect?

gmail — Comment récupérer libellé "Supprimé" dans Gn

Content dated before 2011-04-08 (UTC) is licensed under [CC BY-SA 2.5](#). Content dated from 2011-04-08 up to but not including 2018-05-02 (UTC) is licensed under [CC BY-SA 3.0](#). Content dated on or after 2018-05-02 (UTC) is licensed under [CC BY-SA 4.0](#). | [Privacy](#)